TAMPERE UNIVERSITY OF TECHNOLOGY

Jani Arvola

Sensor Integration to Wireless Sensor Networks

Master of Science Thesis

Examiners: Prof. Timo D. Hämäläinen,
Prof. Marko Hännikäinen
Subject approved by Faculty of Computing and Electrical Engineering Council meeting on 2nd of June 2010

ABSTRACT

TAMPERE UNIVERSITY OF TECHNOLOGY
Master's Degree Programme in Signal Processing and Communications
ARVOLA, Jani Julius : Sensor Integration to Wireless Sensor Networks
Master of Science Thesis, 77 pages
December 2010
Major: Embedded Systems
Examiners: Prof. Timo D. Hämäläinen, Prof. Marko Hännikäinen
Keywords: Wireless Sensor Network, TUTWSN, sensor integration

In computer world, plug-and-play is a commonplace. In general, the sensor manufacturers use various standard physical interfaces in their sensors but they are not compatible in measurement data transfer. IEEE 1451 and ISA-100 standardization is working on common interfaces for industrial applications. The lack of standards in the field of sensor manufacturer require each sensor to be integrated individually to a wireless sensor node.

This thesis presents the design of a sensor integration process for Tampere University of Technology Wireless Sensor Network (TUTWSN). The integration process eases the sensor integration, shortens the required time to integrate a sensor, and minimizes the frequency of errors and the presence of software bugs. The included documentation minimizes the loss of information in multi-partner projects. The integration process consists of five major phases: selecting a sensor, hardware integration testing, integration to a node, integration to a server infrastructure, and testing phases. The focus of this thesis is in the node and its embedded software.

The integration process is tested with five sensors which are integrated to TUTWSN. The sensors are: multipurpose standard industrial signal sensor, air velocity sensor, power meter sensor, radon sensor, and piezoelectric motion detector. Two of the sensors are using I^2C interface, two are using analog-to-digital converter, and one is using interrupts.

Multipurpose standard industrial signal sensor required designing and calibrating the whole sensor and new software components on each level. Air velocity sensor needed a fitter sensor board for an existing transmitter and a new software application. Power meter sensor required designed and calibrating the sensor and a new software application. Piezoelectric motion detector required a new sensor and modification to motion detection application. Calibration is in all the cases done with an existing sensor and the designed sensor are calibrated according to it. The sensor integration takes less than two weeks time on average from selecting the sensor to adopted

wireless sensor network.

TIIVISTELMÄ

TAMPEREEN TEKNILLINEN YLIOPISTO
Signaalinkäsittelyn ja tietoliikennetekniikan koulutusohjelma
ARVOLA, Jani Julius : Anturin integrointi langattomiin sensoriverkkoihin
Diplomityö, 77 sivua
Joulukuu 2010
Pääaine: Sulautetut järjestelmät
Tarkastajat: Prof. Timo D. Hämäläinen, Prof. Marko Hännikäinen
Avainsanat: Langaton sensoriverkko, TUTWSN, anturin integrointi

Tietokonemaailmassa kytke ja käytä on tavallista, mutta yleensä antureiden valmistajat käyttävät useita standardoituja fyysisiä liityntöjä antureissaan, mutta ne eivät ole yhteneviä mittaustiedon siirrossa. IEEE1451 standardoi liitynnän, mutta tätä tukevat anturit eivät yleensä ole yhteensopivia langattomien sensoriverkkojen rajoitusten kanssa. Standardien puute vaatii, että jokainen anturi on integroitava erikseen nodeen, mutta tämä ei vaikuta seuraaviin työvaiheisiin, jotka standardoidaan tässä työssä.

Työ esittää suunnitellun antureiden integrointiprosessin, joka on käytössä Tampereen teknillisessä yliopiston langattomassa sensoriverkossa, TUTWSN. Prosessi helpottaa antureidenn integrointia, lyhentävät integrointiin tarvittavaa aikaa ja minimoi virheiden esiintyneisyyden ja ohjelmistovirheiden määrän. Dokumentointi minimoi tiedon katoamisen yhteistyöprojekteissa. Antureiden integrointiprosessi koostuu viidestä työvaiheesta: anturin valinta, raudan integrointitestaus, nodeen integrointi, integrointi taustajärjestelmään ja testaustyövaiheista. Tämän työn painopiste on nodessa ja sen sulautetussa ohjelmistossa.

Antureiden integrointiprosessi testataan viidellä anturilla, jotka integroidaan TUTWSN-sensoriverkkoon. Nämä anturit ovat: monikäyttöinen standarditeollisuussignaalianturi, ilmavirtausanturi, tehonmittausanturi, radon anturi ja pietsosähköinen liikeanturi. Kaksi näistä antureista käyttävät I^2C -väylää, kaksi käyttävät analogidigitaali-muunninta ja yksi käyttää keskeytyksiä.

Monikäyttöinen standarditeollisuussignaalianturi vaati koko anturin suunnnittelun, kalibroinnin ja lisäksi uudet ohjelmistokomponentit joka tasolle. Ilmavirtausanturi tarvitsi sovituspiirilevyn olemassa olevalle ilmavirtauslähettimelle ja uuden ohjelmistosovelluksen. Tehonmittausanturi vaati anturin suunnittelun, kalibroinnin ja uuden ohjelmistosovelluksen. Pietsosähköinen liikeanturi tarvitsi uuden anturin suunnittelun, toteutuksen ja muutoksen ohjelmistosovellukseen. Jokaisessa tapauksessa kalibrointi tehtiin olemassa olevalla anturilla ja toteutettu anturi kalibroitiin tämän

anturin arvoihin. Anturin integrointi kestää alle kaksi viikkoa keskimäärin anturin valinnasta käyttöönotettuun langattomaan sensoriverkkoon.

PREFACE

This thesis was carried out at Tampere University of Technology in the Department of Computer Systems in 2009-2010 at Design, Applications, Communications, Implementations (DACI) research group.

I would like express my gratitude to my thesis supervisors Prof. Timo D. Hämäläinen and Prof. Marko Hännikäinen for the subject, their guidance, and given possibility to work in the field of WSNs.

I am grateful to Mr. Lasse Määttä, M.Sc, Mr. Jukka Haaramo, M.Sc., and Mr. Arttu Pulli, B.Sc. for their assistance during the thesis. Also, I would like to thank Mr. Markus Rentto, M.Sc, and Mr. Olli Kivelä for their expertise during this work. I would also like to thank my colleagues Mr. Teemu Laukkarinen, M.Sc, Mr. Ville Kaseva, M.Sc, and many others whom with I have worked with during the years at DACI.

I would also like to thank the merry folk of pub Kultainen Apina and its unlimited supply of foreign beers. Also, I thank radio show Korporaatio on Radio Rock for making my mornings more pleasant.

Most of all, I would like to thank my mother Anne, my sister Jenna, my brother Joona, and my girlfriend Sanni for their endless support. Also, I would like to show my gratitude to all my friends for their support.

Tampere, November 2, 2010

Jani Julius Arvola
Insinöörinkatu 49 A 25
33720 Tampere
FINLAND

tel. +358 45 675 84 95
e-mail jani.j.arvola@tut.fi

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

μA	Microampere
ADC	Analog-to-digital converter
APDU	Application Protocol Data Unit
API	Application Programming Interface
APS	ZigBee Application Support sub-layer
Bq	Becquerel
CFM	Cubic Feet Per Minute
Ci	Curie
CRC	Cyclic Redundancy Check
DAC	Digital-to-analog converter
DB	Database
DC	Direct Current
EEPROM	Electrically Erasable Programmable Read-Only Memory
F	Farad
FET	Field-effect transistor
FFD	Full-function device
FTDMA	Frequency and Time Division Multiple Access
GPS	Global Positioning System
HART	Highway Addressable Remote Transducer Protocol
HPL	Hardware Presentation Layer
Hz	Hertz
I	Current
I^2C	Inter-Integrated Circuit
IC	Integrated Circuit

IP	Internet Protocol
IPv6	IP version 6
ISA	International Society of Automation Standard
ISM	Industrial, Scientific and Medical
LR-WPAN	Low-Rate Wireless Personal Area Network
mA	Milliampere
MAC	Medium Access Control
MCU	Microcontroller Unit
MHz	Megahertz
MIPS	Million Instructions Per Second
NG	Gateway Software
NWK	ZigBee Network Layer
OSI	Open System Interconnection
P	Power
PAN	Personal Area Network
PCB	Printed Circuit Board
PdM	Predictive Maintenance
PHY	Physical layer
PIR	Passive Infra-Red
RFD	reduced-function device
RFID	Radio-frequency identification
RTD	Resistive Thermal Device
SMBus	System Management Bus
SPI	Serial Peripheral Interface
SRAM	Static Random Access Memory

STUK	Radiation and Nuclear Safety Authority Finland
Sv	Sievert
TCP	Transmission Control Protocol
TDMA	Time Division Multiple Access
TUTWSN	Tampere University of Technology Wireless Sensor Network
TUTWSNR	Tampere University of Technology Wireless Sensor Network Routing
U	Voltage
UART	Universal asynchronous receiver/transmitter
UI	User Interface
UML	Unified Modeling Language
V	Volt
V_{rms}	Voltage root mean square
WAN	Wide Area Network
WPAN	Wireless Personal Area Network
WSN	Wireless sensor network
ZDO	ZigBee Device Object

1. INTRODUCTION

The number of Wireless Sensor Network (WSN) deployments is increasing and the application space is expanding. The application space of WSNs is ranging from garden [7, 44], to home [17], and to military applications [69]. One thing in common to all deployments and applications is that they need specialized sensors for physical phenomena sensing. Now, when portable device market is growing rapidly, the sensor manufacturers have put on market new sensors with decreased energy consumption for battery powered devices. This has greatly increased the variety of possible sensors to be integrated to WSNs.

1.1 Wireless Sensor Network

The WSN consists of a large number of network devices [1] that are called nodes or motes [28]. Most of the WSNs are used for sensing physical environment and gathering data but also other uses for the WSNs are emerging, such as positioning. The nodes include hardware and embedded software. The nodes are resource-constrained and communicate with each other at ranges from few meters up to kilometers [76] by multi-hop routing. In addition, the nodes should have a battery lifetime of years [1] rather than months to give the network longevity. These add challenges to hardware and software design [36] in wireless sensor network.

The nodes are constrained in computation and processing capabilities, due to the ultra low-power consuming. The nodes are also small in size, which limits the power source to be battery-sized.

The typical WSN is presented in Figure 1.1. The WSNs consist of the network itself and the server infrastructure which is used to store and process data. The WSN can consist of thousands of nodes and therefore nodes should be as cheap as possible. One use case would be that the nodes are disposable and when the node has depleted its power source it is replaced with a new node with new hardware and software.

In wired sensor installations of today, the cable is usually the least expensive component of the installation and labor costs are the highest. By using the WSN, the installation labor costs are reduced because the WSN is easily installed. The goal of the WSN development is to reduce the costs of the nodes even further so that the most expensive component in a sensor installation would be the sensor itself.

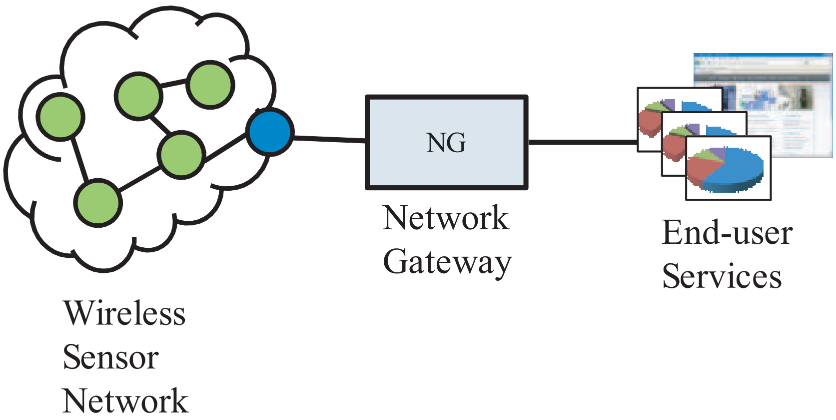

Figure 1.1: *Typical Wireless Sensor Network structure. WSN gathers measurements, forwards them to Network Gateway and the measurements are visualized with end-user services.*

The WSN usually needs a data sink with a gateway device which acts as a bridge between the WSN and other systems. The sink node transfers data through other networks to a user interface or a database, and with the use of WAN, like the Internet, the distance between the user and the WSN can be long. The WSN server infrastructure may implement complex data processing functionality [35] which could not be implemented in the WSN. The server infrastructure may contain a user interface (UI) for data visualization and for controlling the WSN, and a database (DB) for data storing purposes. The WSNs can also be completely autonomous and work without user interference and control and act independently according to measurements.

The wide application space of the WSNs makes it impossible to find a fixed solution for WSN that would work for all of the applications [35]. While IEEE 802.15.4 Low-Rate Wireless Personal Area Networks (LR-WPAN) [71] and ZigBee [4] exist they are not applicable to all potential applications [54].

The commercialization of WSNs has not yet commenced in large scale because of the immature technology and the lack of standards [35]. The most of the WSN products are development kits [26, 42, 51], node platforms with no functionality [20, p.2], or complete monitoring kits [43, 41] but there are no actors on all of the application fields.

1.2 Wireless Sensor Network applications

This section describes some of the WSN applications and sensors needed in them so far. It can be observed that the WSN can vary within the same field of application. Also, to be noted in the applications is that they describe the interfaces used in the sensors. These applications use sensors to measure the same quantity but the sensor interface selected and used vary between analog and digital between the applications.

1.2.1 Industrial monitoring

One industrial application of the WSN is Predictive Maintenance (PdM). PdM or condition-based maintenance attempts to prevent long and costly downtime by predicting when machinery is going to fail and replace it well before. Another application field is to replace wired sensors with wireless ones that brings cost efficiency and allow more dense measurement point coverage.

The example of industrial monitoring is a PdM application, *FabApp* [33]. In FabApp one form of PdM, vibration analysis, is used and a WSN is used to gather vibrations. The WSN used industrial accelerometer connected to analog-to-digital converter (ADC) of a sensor node and vibrations were gathered with high sampling rate of 19.2 kHz. Two nodes gathering vibration data from electric motors are presented in Figure 1.2. One measurement consisting of 3000 data points was obtained by using averaging and windowing of data points. The measurement was then sent to a database through a gateway node. Same network was used onboard on an oil tanker for the same PdM application.

The key motivation in FabApp was to discover which is the cheapest: to gather PdM data with manual collection, online system with 1:1 ratio between the sensors and collection points, or to use a WSN. The cheapest was manual collection which had the highest labor costs. The second cheapest was the WSN which had two times the total cost of manual collection but half the price of online system. Today the cheapest would be the WSN.

Figure 1.2: *Sensor nodes deployed in the FabApp [33].*

1.2.2 Agricultural monitoring

Some cultivated plants, e.g. vines, require stable atmospheric conditions, stable temperature and moisture which need to be within certain range to prevent crop loss and plant damage. However soil temperature is even more important. In dry rural areas water is shared and limited resource and its presence in the soil is also important for cultivated plants.

First, an example of vineyard monitoring WSN is presented in [7]. The vineyard monitoring WSN had thermistors in each node which measured air temperature. Thermistor worked in voltage divider circuit and was connected to an analog-to-digital converter of the node. The WSN consisted of 65 nodes over an area of 0.80 hectares. One of the nodes is presented in Figure 1.3 on the left. This dense network provided temperature measurements with ten minute intervals and the temperature measurements suggests that more money profiting crops could be cultivated in some parts of vineyard, also frost pockets were discovered.

Second, an example of complete WSN system for agricultural monitoring consisting of static and animal-borne nodes is presented in [86]. The static nodes had an onboard temperature sensor and a soil moisture sensor or a camera. The static nodes were solar powered. The soil moisture sensor is capacitance-to-voltage converter and was connected to an ADC. The camera was connected to a digital signal processor which processed the image. The images sent were used to evaluate grass condition in pastures.

The mobile nodes, integrated into a collar, used a different platform with Global Positioning System (GPS), compass, accelerometer and temperature sensors [86]. Only temperature sensor and GPS are digital, and the rest of the sensors needed six ADCs [67]. The positioning data from the GPS combined with accelerometer and compass data provided information about animal grazing and ruminating behavior. Further development added RFID tag reader which was able to read animal ear-tags [68].

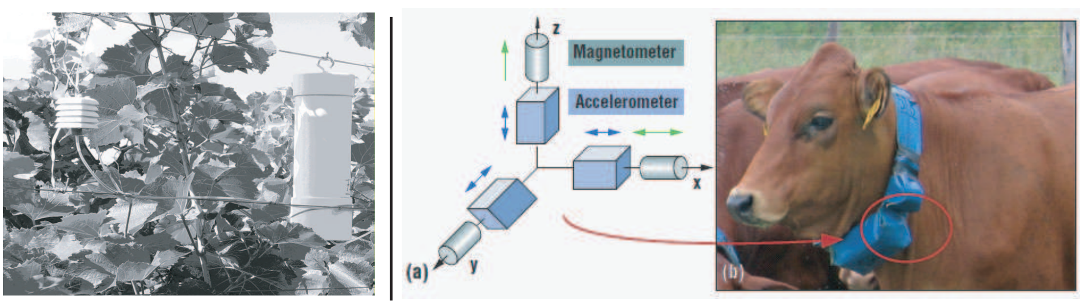

Figure 1.3: *On the left: mote hanging in a vineyard with Stephenson screen [7]. On the right: (a) sensor coordinate system which is fitted into a collar (b) [86].*

1.2.3 Habitat monitoring

Habitat monitoring of animals is quite difficult because the monitoring method should not affect the behavior of the observed animals, thus a WSN with many small nodes is optimum for minimizing the effect of observing method. Two examples of habitat monitoring, first, a WSN capable of recording sounds made by the observed animals [8]. Second, a WSN which continuously measures physical quantities from burrows of birds on *Great Duck Island* [44].

The sound recording WSN is quite simple from the view point of the node: the only sensor needed is a microphone and amplifier but achievements of this application are in other fields, like in remote wake-up module which allows parts of the WSN to be shut down [8].

The Great Duck Island WSN contained two kinds of motes: weather and burrow motes. Weather measurements were gained by using Mica Weather Board which is capable of measuring temperature, illuminance, barometric pressure, humidity and temperature. Only barometric pressure sensor is digital, all other sensors need to be connected to an ADC to function [44]. Burrow measurements on Great Duck Island were achieved using specially designed motes which is presented in Figure 1.4. This mote has a non-contact temperature sensor, an ambient temperature sensor, and a humidity sensor.

Great Duck Island WSN replaced old-fashioned data loggers and provided much more accurate information from the burrows and the habitat of the birds. Also, information would have not been gained without the use of WSN.

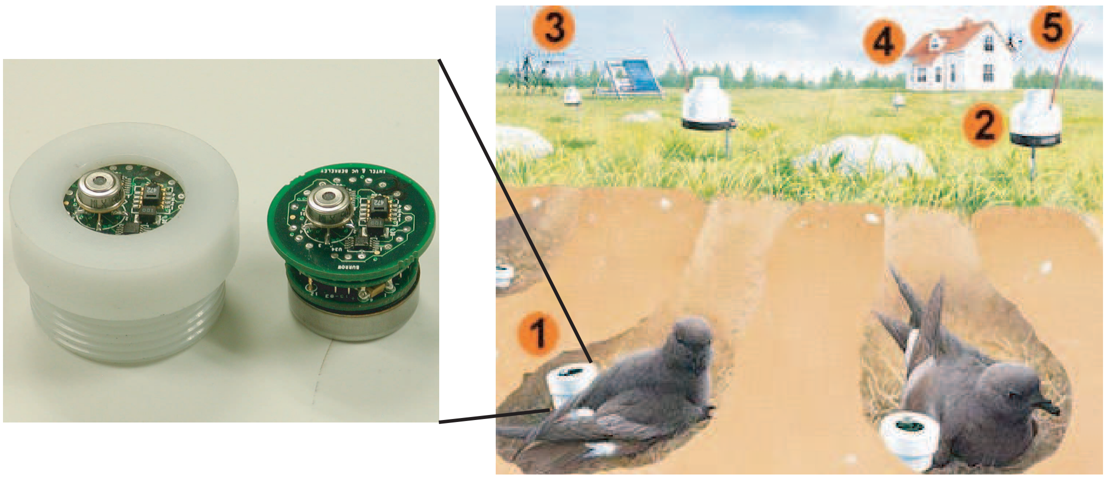

Figure 1.4: *Great Duck Island burrow mote [44].*

1.2.4 Environment monitoring

Using a WSN as an environmental monitoring network vary from monitoring permafrost soil on the Alps [77], *PermaSense*, to seismic monitoring on South American volcano [87]. In PermaSense, the WSN had measurement interval of thirty minutes and a lifetime of several years with special lithium batteries, while the volcano WSN had triggered data acquisition and had a lifetime of approximately one week [2] with two D-cell batteries.

In PermaSense, a custom fibreglass rods were assembled which consisted of multiple temperature, conductivity and moisture sensors, the rods also had individual identification addresses. The rods were then drilled into the mountain rock and were connected to extension boards of nodes which consisted of multiple ADCs [77]. The PermaSense Node sketch is presented in Figure 1.5. The aim of the measurements is to quantify melting processes and heat transportation in the near-surface layer.

The volcano monitoring nodes were fitted with measurement boards which consisted of either a single-axis seismometer or three seismometers aligned in perpendicular fashion. Also, each node was attached to an omnidirectional microphone. All the sensors were analog so they were all connected to ADCs [2]. The goal of volcano monitoring is to explore the potential of WSNs to be used in volcanic field studies.

1.3 Conclusions about Wireless Sensor Network applications

The distributed sensing enables observing a phenomenon more widely than with only one single sensor. A large number of wireless sensor nodes in a specific area

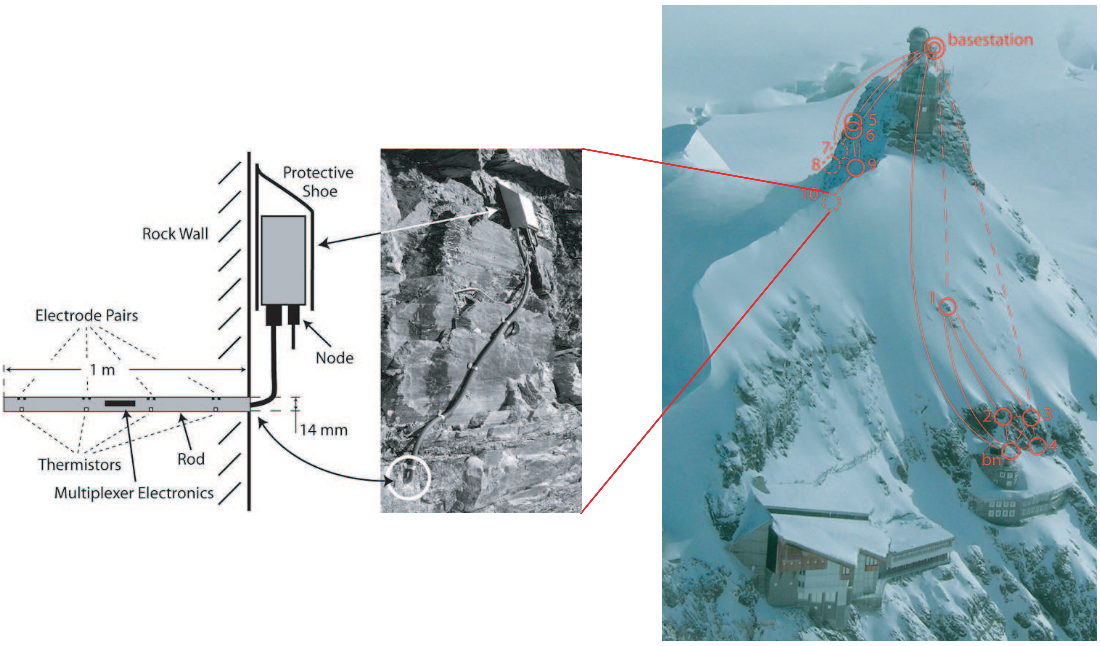

Figure 1.5: *PermaSense deployment on Jungfraujoch [61].*

of interest give a large amount of degrees of freedom to place sensors. Also, the small size of nodes make it possible to gather sensor data in places in which it was previously difficult, if not impossible to obtain. With a WSN which has a long lifetime and self-configuring network topology it is possible to turn network forming to an advantage and place node with special sensor to a place of great interest and place additional nodes which only transfer data.

1.4 The scope of the thesis

To excite also other than engineer minds, the WSN has to provide measurements from interesting phenomena or provide information that reduces costs or is in other ways beneficial. This information is gained with different sensors.

There are many WSN applications but a few of the publications cover the sensor integration to the WSN. Figure 1.6 illustrates this "black box" thinking. What if someone has a measurement need or someone wants a new sensor to the WSN? The focus of this thesis is on the "black box" of the sensor integration. Thesis also discusses the requirements for the sensors to be integrable to WSN and also reveals how to integrate non-WSN suitable sensors.

The WSNs consist of the WSN itself and server infrastructure which both must be taken into account when implementing the sensors to the WSN.

This thesis presents a sensor integration process and it is used on Tampere University of Technology Wireless Sensor Network (TUTWSN). The sensor integration process eases the sensor integration, shortens the time required to integrate a sensor, and minimizes the frequency of errors and the presence of software bugs. The

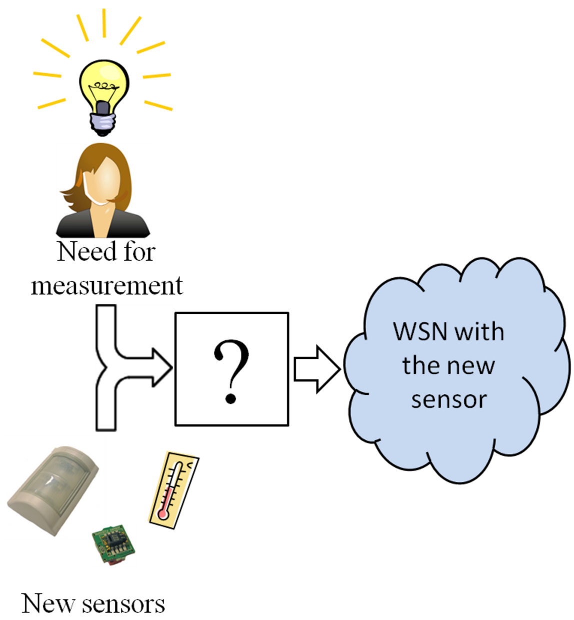

Figure 1.6: *Black box thinking in the sensor integration.*

integration process consists of five major phases: selecting a sensor, hardware integration testing, integration to a node, integration to a server infrastructure, and testing phases. The scope of this thesis is in the node and its embedded software. The integration to the server infrastructure is intended to be introductionary in nature.

The goals of this thesis are: designing a sensor integration process and evaluating the process by using it to integrate sensors, documenting the process, and give an estimate of the sensor integration process duration in TUTWSN.

From the view point of the node there are three major challenges: first, the energy consumption should be as low as possible. Second, the memory consumption should be minimal. Third, designing low-power sensors if there are no sensors for the required quantity that meet the WSN constraints.

The thesis addresses the similarities between TUTWSN and the major WSN standards and thus proving that the integration process which is most likely applicable to other WNSs.

1.5 The outline

This thesis is organized as follows. In Chapter 2, the sensor integration is defined and one promising standard related to sensor integration is presented. Also, Chapter 2 presents the IEEE 802.15.4 Standard which is sometimes referred as the WSN industry standard. Chapter 3 covers TUTWSN that is used to evaluate the integration process. Chapter 3 also addresses the limitations to sensor integration to WSNs. The contribution of thesis to sensor integration is presented in Chapter 4 as the integration process. The integrated sensors which are used to evaluate the process are presented in Chapter 5. Chapter 6 covers time analysis of the integrated sensors. Chapter 7 concludes the thesis.

2. SENSOR INTEGRATION AND WSN STANDARDS

A sensor in this thesis is defined as a device that provides electrical connection to real-world phenomena, acting as a transformer and transforming quantity to measurable signal. The sensor integration is defined as a process in which sensor is integrated to an existing system. In this thesis the system is WSN.

FORCE Technology, SoftNoze USA and many other companies provide sensor integration services but their methods are trade secrets. Their methods are more likely industrial sensor centric [25, 78].

A sensor framework could be one possible tool in the sensor integration. The sensor framework provides a higher level application programming interfaces (API) which are meant to provide consistent method to access sensors in a node. Consistent methods would be ideal but one cannot predict which sensors will be integrated to the device in the future. A good sensor framework would also implement methods needed by the future sensors, not only limiting to pre-existing sensors.

2.1 IEEE 1451 Smart Transducer Interface Standards for Sensors and Actuators

The IEEE 1451 Smart Transducer Interface Standards for Sensors and Actuators [37] is a promising standard in sensor integration. Transducers, defined in The IEEE1451 as sensors or actuators, are devices that convert one type of energy to another. A transducer acts as an interface between a microprocessor and physical world and supports some form of hardware input and output from the microprocessor to the transducer [37]. Many sensor interface and field bus implementations are available, each with its own advantages and disadvantages in the targeted market segments of sensors [14] but with multitude of interfaces, all of these cannot be easily integrated to one system. Therefore, the standard IEEE 1451 was developed to address these issues [14].

The purpose of the IEEE 1451 is to define a set of common interfaces for connecting transducers to microprocessor-based systems [37] and through standardization make it easier for the transducer manufacturers to develop smart devices interfaceable to the networks and the systems [14].

From the view point of the this thesis, the most interesting part of the IEEE1451 standard is IEEE 1451.2 Transducer to Microprocessor Communication Interface. The IEEE 1451.2 defines a Transducer Electronic Data Sheet (TEDS) and modified Serial Peripheral Interface (SPI) for data transfer. TEDS is stored in the memory of the transducer. The TEDS contains fields that describe the type, attributes, operation and calibration of the transducer [59]. With the TEDS feature, changing, upgrading, and replacing transducers are simply ‘‘plug and play’’.

Unfortunately there are not many TEDS sensor manufacturers available. Large automation instrumentation manufacturers Honeywell and National Instruments have a few models for basic quantities but the sensors are too energy hungry to be operated on a battery powered node [24, 23].

There are public, suitable processes for sensor integration to old industries: automotive and automation industry but there are no process for WSNs and that is why a new process is developed and used in this thesis.

2.2 IEEE 802.15.4 Standard

The IEEE 802.15.4 Standard [71] describes physical (PHY) and wireless Medium Access Control (MAC) layers for low-rate wireless personal area networks (LR-WPANs) which operate in 2.4 GHz industrial, scientific and medical (ISM) band. It describes LR-WPAN as a simple, low-cost communication network that allows wireless connectivity in applications. IEEE 802.15.4 is the industry standard for WSNs.

Two different types of devices can participate in an IEEE 802.15.4 network: a full-function device (FFD) and a reduced-function device (RFD). The FFD has three operating modes as a personal area network (PAN) coordinator, a coordinator or as a device. An FFD can communicate with RFDs or other FFDs. An RFD is intended for simple applications that do not require large amount of data transmissions and can only be associated with a single FFD at a time. [71]

IEEE 802.15.4 LR-WPAN is formed around the overall PAN coordinator that forms the initiation and termination point of the network and only one of them can exist in the network. The overall PAN coordinator may have greater computational resources than other coordinators which makes the network hardware heterogenous. The PAN coordinator is the primary controller of the PAN. The PAN coordinator is mains powered and the devices are battery powered. PAN coordinators are capable of routing data. Multiple PAN coordinators provide synchronization services to other devices or other coordinators and area of the network can be increased. [71] IEEE 802.15.4-2006 LR-WPAN can be seen in 2.1. Coordinators make routing decisions, coordinate the network functions and allow devices to join the network or reject them.

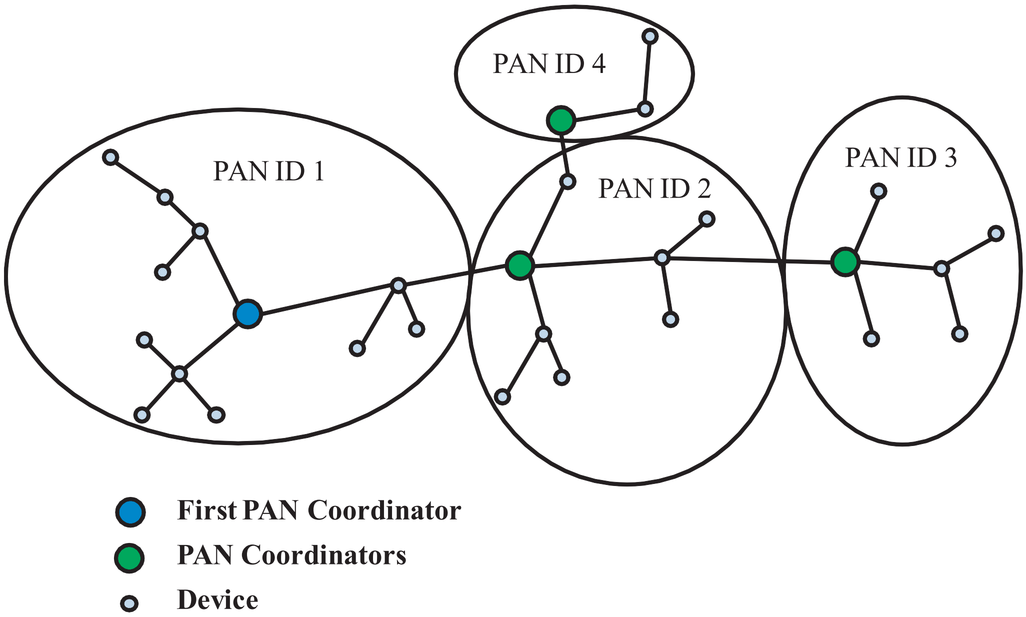

Figure 2.1: *IEEE 802.15.4-2006 cluster tree network [71]*

2.3 IEEE 802.15.4 Compliant Wireless Sensor Networks

The IEEE 802.15.4 compliant radio traceivers ease the WSN platform design and three WSNs based on those are presented: ZigBee,Wireless Highway Addressable Remote Transducer Protocol (WirelessHART), and International Society of Automation Standard (ISA) ISA-100.11a-2009. According to Tekes Nordic WSN Market study [79] these chosen IEEE 802.15.4 compliant WSNs form major part of the WSN market in the Nordic countries.

2.3.1 ZigBee

ZigBee is a specification for low-cost, low-energy wireless communication and it defines application and network (NWK) layers which operate on top of IEEE 802.15.4 defined MAC and PHY layers [4], as depicted in Figure 2.2. The NWK layer provides functionality to MAC layer and interfaces to application layer [4] by providing two services: data transmission service and management service. The data transmission service allows the applications to transport application protocol data units (APDU) between devices and the management service allows the applications to interact with the stack [4]. The ZigBee protocol stack also has Security Service Provider and ZigBee device object (ZDO) Management Plane.

The ZigBee application layer contains the application support sub-layer (APS), ZDO and application objects. The APS provides an interface between NWK and the application layer through a general set of services. The application objects are hosted on ZigBee devices in an environment called *application framework*, and the

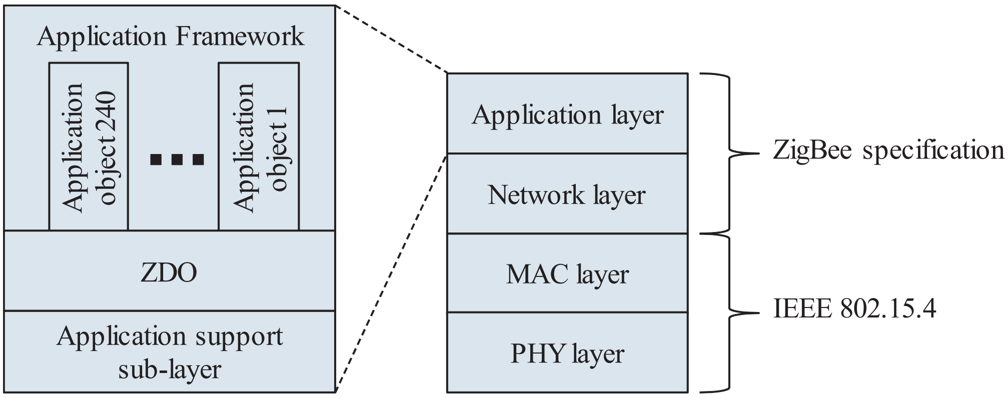

Figure 2.2: *ZigBee Stack Architecture*

environment can host up to 240 application objects. The ZDO is located between the application framework and the APS and it presents public interfaces between the application objects in the application framework and the APS. [4]

2.3.2 WirelessHART

Process monitoring, control and industrial automation have advanced greatly within the last three decades, and field buses, such as Highway Addressable Remote Transducer Protocol (HART), have been available since the 1980s. The HART Communication Protocol is global standard in the field of industrial automation. HART is a digital bus and the digital data is simultaneously communicated with a 4-20 mA analog signal by modulating the HART digital signal in it. The analog signal can be used in older systems.

WirelessHART is a wireless extension to HART protocol. WirelessHART was designed to provide wireless communication to HART protocol and to enable easy retrofitting of wired HART devices to wireless. WirelessHART uses the standard HART Application Layer [18] so WirelessHART only provides network layer on top of IEEE 802.15.4 MAC and PHY layers, this is presented in Figure 2.3.

HART application layer is based on master sent commands with requests for spe-

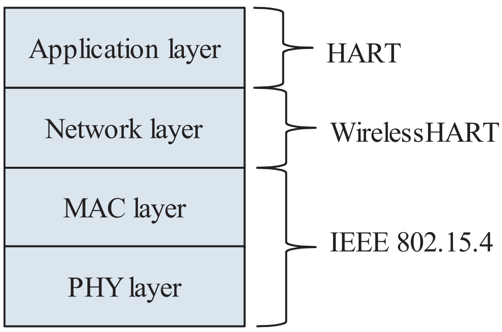

Figure 2.3: *WirelessHART protocol stack*

cific values. Commands are standard data types and provide standard and device-specific status which provide quality assessment and status information for all process variables [18]. Several new features were added to HART to better suit for wireless communication: Smart Data publishing enables data generation only when needed, data samples are time stamped to measure latency. Measurements can also be triggered [18].

2.3.3 ISA-100.11a-2009

ISA-100.11a-2009 is an ISA standard for WSN. It describes WSN as wireless industrial sensor network (WISN). ISA-100.11a-2009 reference model is presented in Figure 2.4. The standard uses MAC and PHY layers of IEEE 802.15.4 which enables the use of IEEE 802.15.4 radio transceivers. The network layer of ISA standard uses headers which are compatible with the Internet Engineering Task Forces 6LoWPAN standard [57] and the transport layer offers connectionless service over IP version 6 (IPv6) which offers security, authentication and encryption.

The application layer defines software objects to model real-world objects, and also defines the communication services necessary to enable object-to-object communication between distributed applications [57]. The application layer consists of an application sub-layer which communicates with user application objects through application sub-layer data entity service access point (ASLDE-n SAP). There is one-to-one relationship between ASLDE-n SAPs and user application objects. ASLDE-0 SAP is reserved for device management application process which is dedicated to manage the standard-compliant device and its communication services [57].

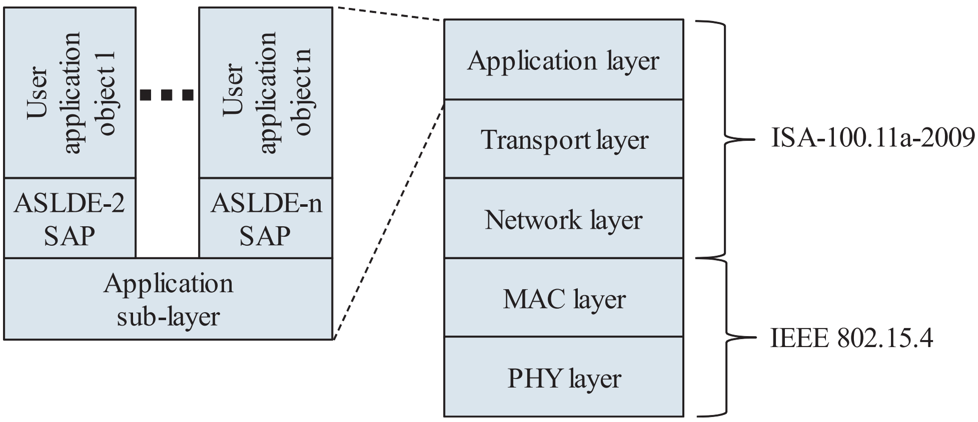

Figure 2.4: *ISA protocol stack*

2.4 Sensor integration to IEEE 802.15.4 Compliant Wireless Sensor Networks

All of the presented IEEE 802.15.4 compliant WSNs define an application layer. The application layer is used by the applications to utilize data transferring capabilities of lower layers.

An application can be used to read sensors values and transfer them through WSN to a server infrastructure. The application layer in each case sets limitations to resource and execution time of an application which must be taken in to account when implementing a new sensor application. The deeper exploration and integrating a sensor to these networks is left to future work of this thesis.

2.5 Sensor Boards Available for Wireless Sensor Networks

Libelium Comunicaciones Distribuidas S.L. [70] is a manufacturer of ZigBee compatible motes which are called WASP motes. The WASP motes do not have sensing capabilities and the capability is gained with various sensor boards. The boards are Gases Board [40], Events Board [39], and Agriculture Board [38]. The sensors available in the sensor boards are presented in Table 2.1.

The Gases Board is capable of measuring 6 different gasses and humidity. All of the sensors in the Gasses Board are analog and require analog-to-digital converter from the WASP mote.

The Events Board allows simultaneous use of 8 different motion detector sensors. The sensors are either analog output or switch type sensors. The analog sensors are connected to the analog-to-digital converter of the WASP mote. The switch type sensors are connected to the interrupt input of the WASP mote.

The Agriculture Board supports up to 14 sensors at the same time. The Agriculture Board is the only sensor board from Libelium which has a digital sensor. The only digital sensor is a humidity sensor and all the other sensors are analog.

MEMSIC Corporation [10], previously known as Crossbow, manufactures IEEE/ZigBee 802.15.4 compatible MICA2 [47] and TelosB [50] motes. External sensor boards need to be used with MEMSIC motes because they lack sensing capabilities. Four different sensor boards are available from MEMSIC: MTS100-series, MTS300-series, MTS400-series and MTS510CA-series. All of the sensors in each board is presented in Table 2.1.

The MTS100-series sensor board has a thermistor which is used to measure temperature and a light sensor. The both of the sensors need to be connected to the analog-to-digital converter of the mote.

The MTS300-series sensor board has the same thermistors and the same light sensors as the MTS100-series. The MTS300-series sensor board also has a microphone and a 2-axis accelerometer depending on the model. The microphone and the 2-axis accelerometer are both analog and are connected to the motes analog-to-digital converter.

The MTS400-series sensor board has a combined humidity and temperature sensor, a combined atmospheric pressure and temperature sensor, a light sensor, and a 2-axis accelerometer. The only analog sensor in MTS400-series sensor board is the 2-axis accelerometer. The combined humidity and temperature sensor is connected to the mote with 2-wire interface. The combined atmospheric pressure and temperature sensor uses a 3-wire interface and the light sensor uses System Management Bus (SMBus).

The MST500-series sensor board has a microphone, a light sensor and a 2-axis accelerometer. All of the sensors in the MTS500-series sensor board are connected to the analog input interface of the mote.

TinyNode [83] is a product of Shockfish SA [64]. TinyNode needs an external sensor board to sense and that is why Shockfish has manufactured TinyNode Standard Extension Board [82]. The TinyNode Standard Extension Board has a light sensor, a temperature sensor and a humidity sensor. The humidity sensor is the only sensor providing digital interface. The digital interface in the humidity sensor is 2-wire interface.

Table 2.1: *Measurement quantities in commercially available sensor boards.*

Quantity	**Libelium Agri. Board [38]**	**Libelium Gases Board [40]**	**Libelium Events Board [39]**	**Crossbow MTS100 [11]**	**Crossbow MTS101 [11]**	**MEMSIC MTS300 [48]**	**MEMSIC MTS310 [48]**	**MEMSIC MTS400 [49]**	**TinyNode Ext. Board [82]**	**Crossbow MTS510 [11]**
Temperature	x			x	x	x	x	x	x	
Humidity	x							x	x	
Soil moisture	x									
Soil temperature	x									
Leaf wetness	x									
Atmospheric pressure	x							x		
Solar radiation	x									
Ultraviolet radiation	x									
Anemometer	x									
Gas - CO		x								
Gas - CO_2		x								
Gas - O_2		x								
Gas - CH_4		x								
Gas - H_2		x								
Gas - NH_3		x								
Gas - H_2S		x								
Gas - NO_2		x								
Pressure			x							
Bend / Stretch			x							
Vibration / Impact			x							
Luminosity			x	x	x	x	x	x	x	x
Passive infrared			x							
Switch			x							
Sound pressure			x			x	x			x
Acceleration							x	x		x

3. TAMPERE UNIVERSITY OF TECHNOLOGY WIRELESS SENSOR NETWORK

This chapter describes Tampere University of Technology Wireless Sensor Network (TUTWSN) which has been developed in the department of computer systems in the DACI research group.

The TUTWSN is a *sense-and-send* WSN but it has functions varying from sensor to actor network within the same network. The design principle behind TUTWSN is that all nodes are self-configuring and form the network without the need of a coordinator or outside coordination.

The TUTWSN has three node roles: *subnode*, *headnode* and *gateway node*. Subnodes are measurement centric members, headnodes have the same measurement abilities as the subnodes but are also capable of routing data, and gateway nodes act as gateways between other networks and TUTWSN. Also, the gateway nodes propagate interests to the network. TUTWSN forms clusters around routing headnodes and non-routing subnodes. Headnodes route data to other headnodes and finally to a gateway node. A similar cluster tree as in IEEE 802.15.4 Chapter 3 is presented in figure 3.1 with roles changed to TUTWSN node roles.

The TUTWSN nodes are identical in hardware which makes the network hard-

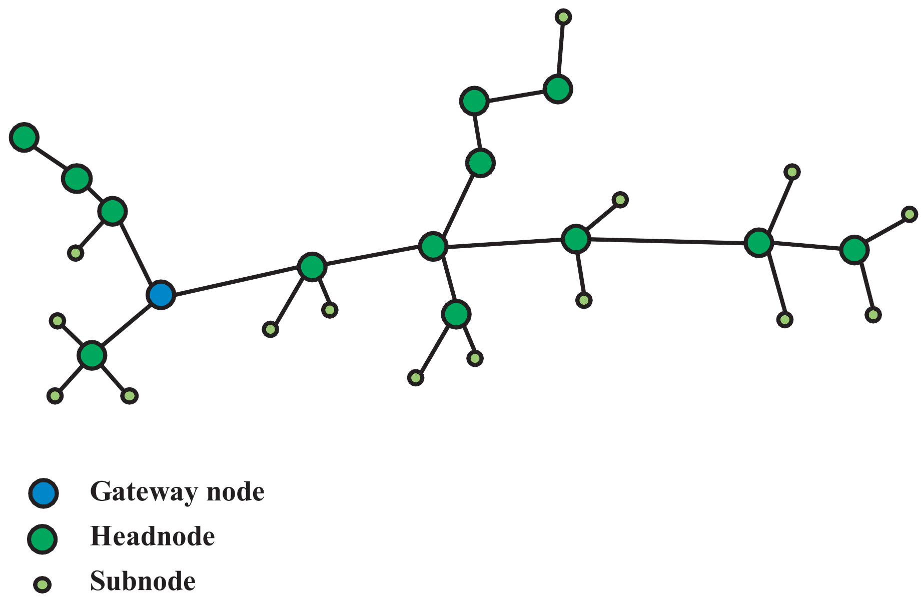

Figure 3.1: *TUTWSN cluster tree network*

ware homogenous. The gateway node differs in hardware because it has parts that provide gateway properties. This design principle decreases the costs of the network.

The TUTWSN is not only a WSN but it also includes the server infrastructure in the form of a database, a gateway software component, and a user interface. Plug-in architecture makes it possible to expand the server infrastructure with new software components.

3.1 TUTWSN Protocol Stack

The TUTWSN protocol stack contains a MAC, multi-hop routing, middleware and application layers [76]. The protocol stack is presented in Figure 3.2. The key layer for WSN network formation, channel access, and power management is the MAC layer [34]. A routing protocol is responsible for creating multi-hop routes which enable end-to-end communcation [34]. A middleware layer hides the underlying protocol stack and node platform from the application layer and applications.

3.2 TUTWSN Medium Access Control Layer (TUTWSN MAC)

WSNs require an energy efficient MAC protocol that is able to minimize the radio active time [30]. As mentioned before the radio is the most power consuming component and its activity must be kept as low as possible.

The MAC layer operates on top of the physical layer and is a sublayer of the Data Link Layer in the Open System Interconnection (OSI) model [32]. The MAC layer manages radio receptions and transmission and therefore has a great effect on the energy consumption of node and on network performance [31], [32].

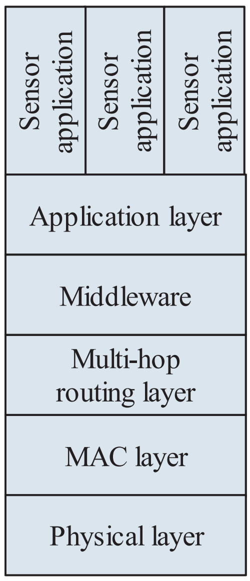

Figure 3.2: *TUTWSN protocol stack*

The TUTWSN MAC uses Frequency and Time Division Multiple Access (FT-DMA) based channel access. The TUTWSN MAC utilizes frequency division, *cluster channels*, between different clusters and time division within a cluster. Also a network-global signaling channel, *network channel*, is used for advertising and detecting clusters by listening for beacons [30].

Superframe-based data exchange scheme, seen in Figure 3.3, is maintained by each cluster head on a cluster channel. Communication between nodes is executed within the superframes which are generated in fixed intervals (access cycle time). Between the superframes cluster heads communicate with other cluster heads, send and receive network beacons and sleep. Within the superframe data exchange is executed in communication slots: reserved and ALOHA slots. Data is exchanged in collision free reserved slots while ALOHA slots are used when joining a cluster and requesting a reserved slot. Each superframe begins with a cluster beacon which contains slot information.

The sleep time between superframes can alternatively be used to read sensors and calculate measurements but it is compulsory for nodes to take part in each neighboring superframes of headnodes to maintain reliable links. Within the same area superframes do not over lap and maximum superframes in one access cycle depends on the superframe interval.

3.3 TUTWSN Routing Layer (TUTWSNR)

Traditional routing protocols do not take into account that the energy supply is limited and a node contains only a limited power supply [66]. That is why WSNs need specially designed routing protocols.

TUTWSNR is an energy-efficient multi-hop routing protocol which uses cost metrics to create gradients *routes* from one node to another [75]. The cost metrics are combined to a single value with a cost function and each of those metrics are

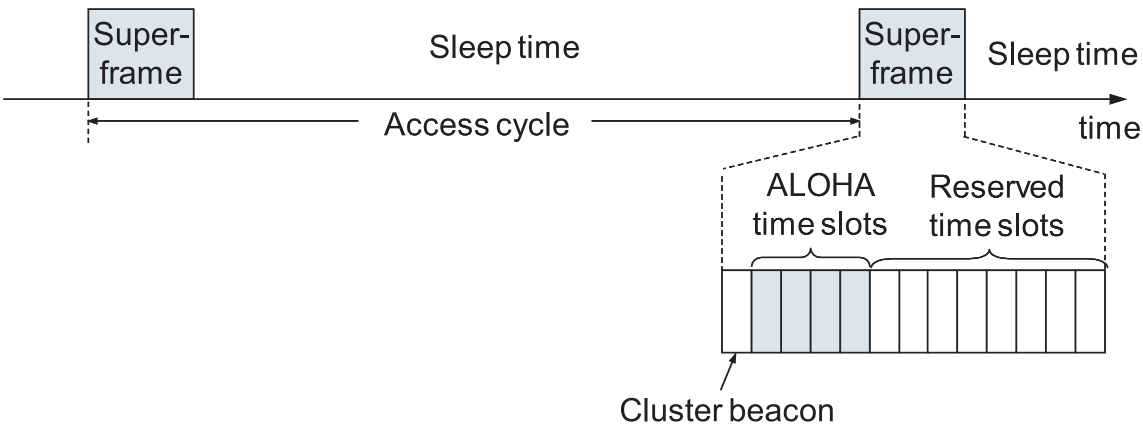

Figure 3.3: *TUTWSN access cycle and superframe [31].*

weighted with scaling factor [75]. At first, when TUTWSN is powered, the gateway node floods its presence into the network by route advertisements (RADV) as shown in Figure 3.4. Headnodes that receive the advertisements create routes to the gateway node and calculate the costs of those routes. If cost decreases, the nodes transmit new costs in RADV to ther neighbors. Next, neighboring nodes that cannot hear the gateway query for routes. When routes are requested, the node queries RADV from the node that has lowest cost. When the network is formed and a new packet is received, the node transmits the packet to its neighbor and towards the gateway node. The gateway node requests data from the network by sending interest advertisements (IADV) in the reverse direction of the routing gradients. A node forwards IADV if it is new otherwise discards it.

The cost metrics are combined to a single value with a cost function and each of those metrics are weighted with scaling factor [75].

A cross-layer designing between TUTWSN MAC and TUTWSNR routing layers provides better performance: cluster beacons contain routing information and on the other hand route advertisement packets include channel and TDMA timing information.

3.4 TUTWSN Application Layer

TUTWSN application layer contains sensing applications for each sensor application, actuator applications, diagnostics applications, and network configuration applications which form several parallel processes [36]. Each application is timed according to the network interests to execute with accurate interval, normal interval being two minutes. After the interval, application is executed. The application constructs a radio packet and hands it to lower layer.

The TUTWSN data packet structure is presented in Figure 3.5. 12 bytes header is preserved for TUTWSN functionality and it contains information related to lower layers in the TUTWSN protocol stack. The rest of the payload, 20 bytes, is usable in applications. From the 20 bytes, 2 bytes are used identically in each application.

Application header contains two bit application layer version number and 5 bits

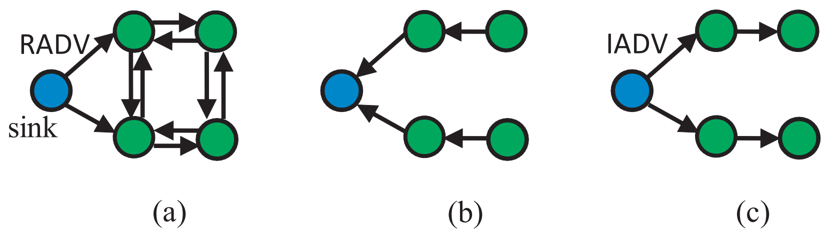

Figure 3.4: *Gateway node initiated route formation [75]. a) Gateway advertises its presence with RADV. b) Nodes establish routes towards the gateway node c) Gateway node broadcasts interests*

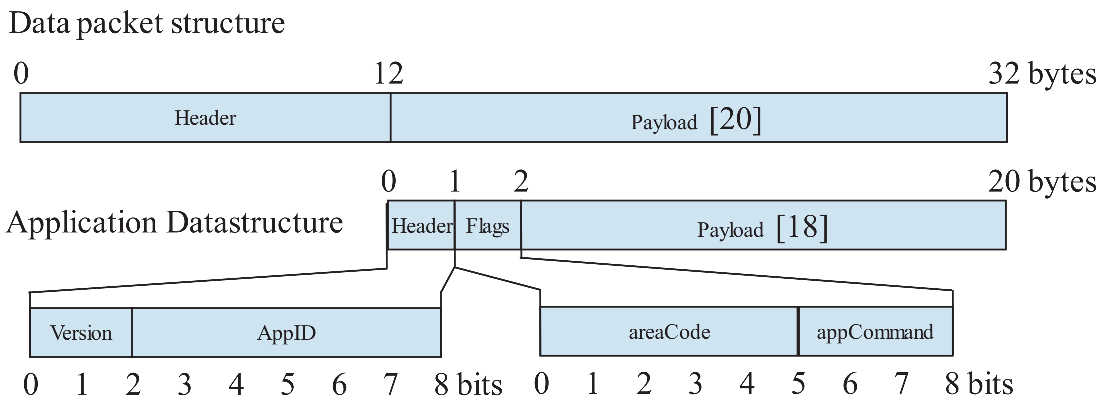

Figure 3.5: *TUTWSN data packet structure and application Data structure.*

of application specific identifier, AppID. AppID is used at a Network Gateway to process the payload accordingly but it can also be used at the nodes to react or process measurements from other nodes.

The flag byte contains 5 bits areaCode and 3 bits appCommand. areaCode can be used to switch TUTWSN between node and data centric modes in the network itself and appCommand can be used to send simple commands between the nodes.

The remaining 18 bytes of the payload are usable by the application.

3.5 TUTWSN Hardware Prototypes

The DACI research group has developed hardware prototypes for wireless sensor networks for over six years [29]. Deeper TUTWSN hardware prototype design principles can be reviewed from [31] but major design principle has been that all the components are commercially off-the-shelf (COTS) components. Implementing and testing has been conducted in varying environments and TUTWSN nodes are CE approved. Typical block diagram of a TUTWSN node is presented in 3.6. TUTWSN node consists of four subsystems: *power*, *sensing*, *computing* and *communication* subsystems and each of them bring limitations to sensing subsystem.

In power subsystem, electric power is usually generated from batteries but in

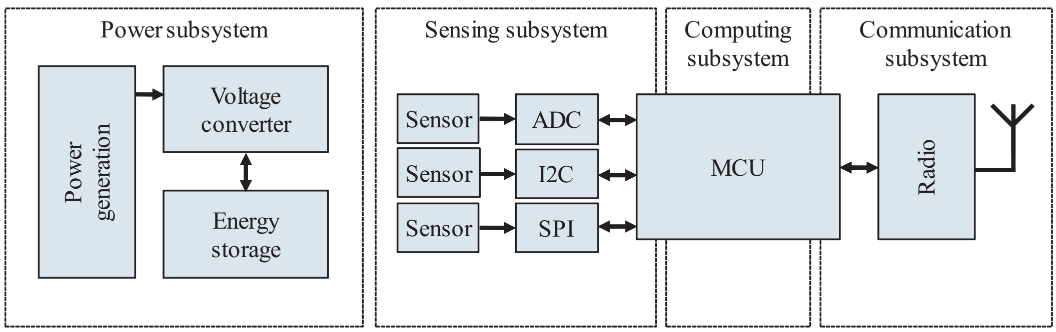

Figure 3.6: *Block diagram of the TUTWSN node.*

outdoors solar cells with rechargeable batteries have been used. Also battery voltage is regulated to known operating voltage using a regulator to provide stable supply voltage to the node.

Gateway nodes need to be constantly on to keep the Transmission Control Protocol/Internet Protocol (TCP/IP) connection alive and to transfer data to other systems or databases. To meet this increased energy consumption they are connected to 5 volt DC wall adapter power supply. TUTWSN nodes are identical and all of them are assembled with same DC-connector to keep voltage drop and power consumption in the regulator as low as possible, MAX8880 [45] regulator is used. The MAX8880 does not provide high current and limits maximum current to 200 mA. Current peaks of the node are almost 20 mA and this leaves 180 mA to sensors. Voltage of the node is low, 2.50 volts which enables them to be battery powered but also limits the variety of sensors that meet the voltage reguirement.

Computing subsystem limits the raw sensor data processing capabilities. TUTWSN nodes use Microchip PIC18LF8722 Microcontroller Unit (MCU) [52] which runs at 4 MHz and provides 1 MIPS. Processing power is not a big issue because processing can be distributed among the nodes and background system.

The issue is memory consumption. PICs use the Harvard architecture which separates program and data memories. The flash program memory is 128 kilobytes and data memory consists of 3936 bytes of Static Random Access Memory (SRAM). 1024 bytes of Electrically Erasable Programmable Read-Only Memory (EEPROM) exists but it is reserved for static data.

The communication subsystem consists of a radio, antenna and an interface between the MCU and the radio. The communication subsystem limits the amount of sensor data which can be wirelessly transmitted. TUTWSN nodes use Nordic Semiconductor nRF24L01+ [56] radio which limits maximum radio packet payload size to 32 bytes. This radio offers an internal Cyclic Redundancy Check (CRC) calculation which enable retransmissions to be done on physical layer. Also, 433 MHz version with nRF905 [55] radio exists. A printed circuit board antenna is used to lower costs [21].

The sensing subsystem consists of various sensors and actuators which are connected to interfaces of the MCU. Three serial interfaces are provided by the MCU: Inter-Integrated Circuit (I^2C), Serial Peripheral Interface (SPI) and Universal asynchronous receiver/transmitter (UART). Also many IO-pins of the MCU are available.

A real life implementation of TUTWSN node, called 2.4 GHz universal node, is presented in 3.7. This node operates on 2.4 GHz band. The node is transformed to a gateway node with an Ethernet module, this version is seen in Figure 3.7 on the left.

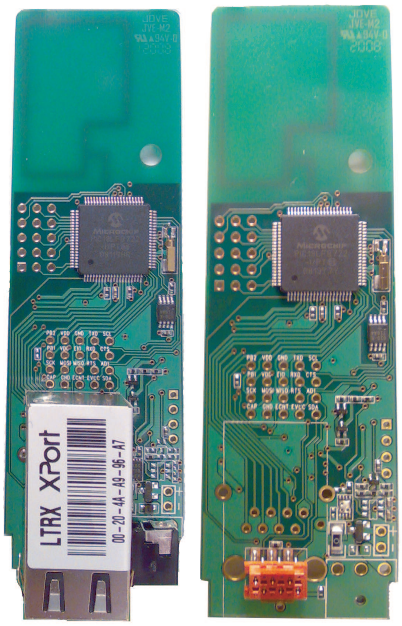

Figure 3.7: *On the left TUTWSN universal node with Ethernet module and on the right node without the module.*

The TUTWSN nodes have enclosures which protect nodes for distribution, storage, sale and from environmental hazards. Enclosure is also needed for an electric product to receive CE approval. Enclosure must be made of material which does not interfere with radio frequencies and it must be thick enough to withstand static electricity shocks. The enclosure can be one of the sensor selecting parameters.

The TUTWSN nodes run algorithms which consume large amounts of memory which can be seen in table 3.2. The greatest amount of memory is consumed by algorithms that make the WSN a reality, memory consumption by layer is presented in table 3.1. Also, in TUTWSN deployments a basic set of sensor applications are used in all nodes. The basic set of sensors consists of temperature and illuminance applications.

A large number of low power and suitable sensors exits for WSNs. For exam-

Table 3.1: *Memory consumption of the TUTWSN node by layer.*

Layer	Program bytes	Data bytes	Program %	Data %
MAC	34 562	570	26.4	13.9
Management	17 252	380	13.2	9.3
Routing	14 974	320	11.4	7.8
Misc	13 070	1650	10.0	40.3
HW	11 024	230	8.4	5.6
Application	8 520	347	3.0	8.5
Total	8 520	347	3.0	8.5

Table 3.2: *Memory consumption table of the TUTWSN node by application.*

Target	Program bytes	Data bytes	Program %	Data %
Node	81 610	3488	62.3	88.6
Headnode	78 908	3482	60.2	88.4
Subnode	64 389	3154	49.1	80.1
Gateway node	86 325	3660	65.9	92.9

Application	Program bytes	Data bytes	Program %	Data %
Acceleration	5936	10	4.5	0.3
Acceleration v2	7199	39	5.5	1.0
Activity monitor	1745	8	1.3	0.2
Capacitance	2101	30	1.6	0.8
CO_2	485	2	0.4	0.1
Float	1085	27	0.8	0.7
Magnet switch	1073	27	0.8	0.7
Multitemp	2227	33	1.7	0.8
Photo diode	671	4	0.5	0.1
Piezo	1147	29	0.9	0.7
PIR	1145	21	0.9	0.5
Pin monitor	1450	25	1.1	0.6
Power meter	2035	12	1.6	0.6
Soil moisture	591	2	0.5	0.1
Temperature	2322	20	1.8	0.5

Misc	Program bytes	Data bytes	Program %	Data %
Autoconfig	3143	1	2.4	0.1
Bootloader	5353	15	14.1	0.4
Diagnostics	1267	13	1.0	0.3
Global time	3610	68	2.8	1.7
Neighbors	1350	4	1.0	0.1
Path	664	4	0.5	0.1
Route test	539	5	0.4	0.1
SW Adv	1425	7	1.1	0.2

ple: acceleration, humidity, illumination and temperature which form basic set of sensors in TUTWSN, these sensors are on the same circuit board as MCU but additional sensors must be connected to node otherwise. Important requirements for sensors are low power consumption and quick measurement event, which are major determinants of the sensors energy consumption [31]. The features of some sensors integrated to TUTWSN are presented in Table 3.3. All of the sensors in Table 3.3 fulfil the requirements.

Table 3.3: *Features of typical sensors [31, p.27]*

Physical quantity	Example sensor	Accuracy	Active current	Sensing time	Energy cons.
Acceleration	VTI SCA3000	1%	120 μA	10 ms	3.6 μJ
Air pressure	VTI SCP1000	150 Pa	25 μA	110 ms	8.3 μJ
Humidity	Sensirion SHT15	2%	300 μA	210 ms	190 μJ
Illumination	Avago APDS-9002	50%	2.0 mA	1.0 ms	6.0 μJ
Infra-red	Fuji MS-320	-	35 μA	cont.	-
Magnetic field	Hitachi HM55B	5 %	9.0 mA	30 ms	810 μJ
Soil moisture [62]	Decagon EC-5	3%	2.0 mA	10 ms	50 μJ
Temperature	Dallas DS620U	0.5°C	800 μA	200 ms	480 μJ

3.6 TUTWSN server infrastructure

In addition to the WSN, TUTWSN also consists of Network Gateway software (NG), Sensor Network Application Platform (SNAP), measurement database, and Monitoring user interface. These form TUTWSN backbone architecture. Currently the Monitoring user interface is a WSN Control Panel which is mainly used for development purposes. All of the TUTWSN components are presented in Figure 3.8.

Gateway node has a predefined server IP address and when powered on, it connects to the server. From the server it looks for appropriate NG address and tries to connect to it. NG then handshakes with gateway node and confirms that the node ID of the gateway node and radio address are legal. If the previous step fails the NG

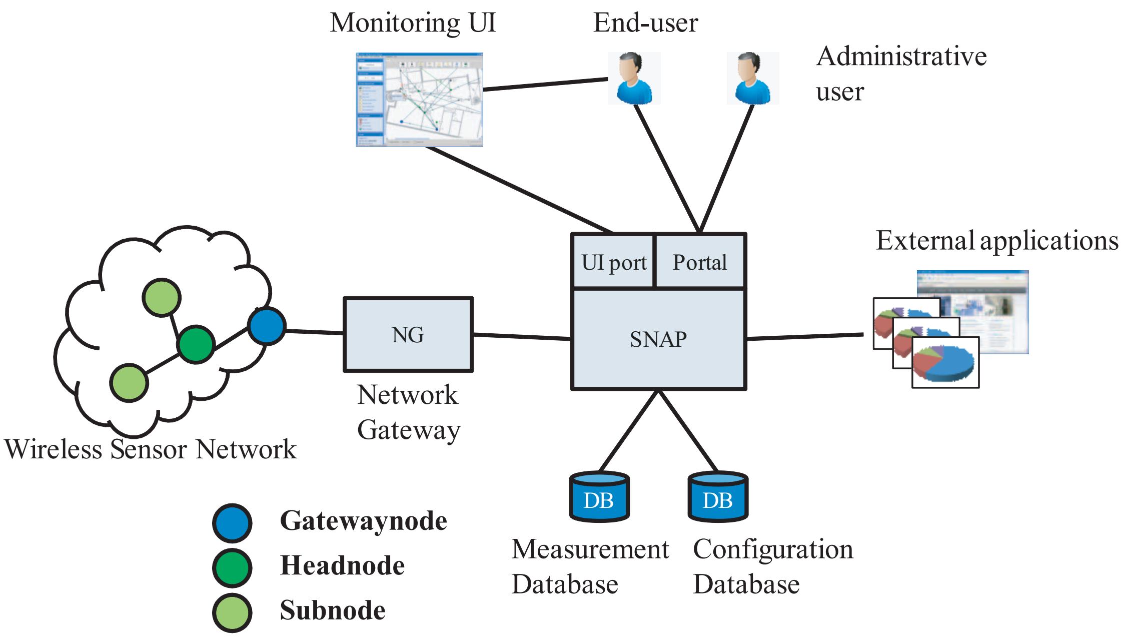

Figure 3.8: *TUTWSN server infrastructure*

disconnects the gateway node. After successful handshake NG gives the gateway node interests from configuration database. According to these interests the WSN starts to gather data which is directed through NG to SNAP, which stores data to a measurement database, user interfaces and external applications.

SNAP forms an integral part of centralized WSN infrastructure. SNAP provides functionalities to applications as services and also enables messaging between applications. [27] The major benefit of SNAP is that it transforms deployment centric infrastructure with multiple scattered DBs, NGs, WSN Control Panels, and configurations, to centralized infrastructure with only one communication point which is SNAP itself.

Network Gateway acts as a gateway between the gateway node and the SNAP. The gateway node packs WSN data to IP packets and sends them to NG which unpacks IP packets and hands the WSN data to SNAP through Data Processing Applications. Network Gateway has to know all the packet types and all the others are rejected and put to discarded packets.

WSN Control Panel is a user interface which enables WSN controlling and provides measurement data visualization techniques. WSN Control Panel is used with TUTWSN but it is possible to use WSN Control Panel with any WSN when using SNAP.

WSN Control Panel shows measurement values in real time and past values. The real time data visualization has two possibilities in WSN Control Panel: in measurements a value is visualized in a box next to the node icon presented in Figure 3.9, and in alert event blinking triangle is drawn, seen in Figure 3.10.

WSN Control Panel needs to connect to a DB to visualize measurements from the past in graph form which is presented in Figure 3.9. Time frame can be set to cover the nodes entire life-time in one network. Also, latest value is visualized next to the node icon, see Figure 3.9, which is which is visualized when new value arrives. In alert packet approach only new bit handling has to be added. Alert packet visualization is done with an alert dialog, visualized in Figure3.10. The alert triangle blinks next to node icon expanding to the alert dialog which is presented on the left in Figure 3.10.

Past events are visualized in graphical form as presented in Figure 3.9. WSN Control Panel also provides access to WSN diagnostics data visualization.

WSN Control Panel is used to control interests and measurement interval of WSN and also to control WSN actuators. Interests contain information on what data nodes should gather if the nodes have the needed sensors.

WSN Control Panel is used to set and modify SMS and E-mail alarms. The alarms are located in the SNAP which handles alarm message sending. Node centric alarms trigger when measurement value of node undershoots or exceeds the alarm condition.

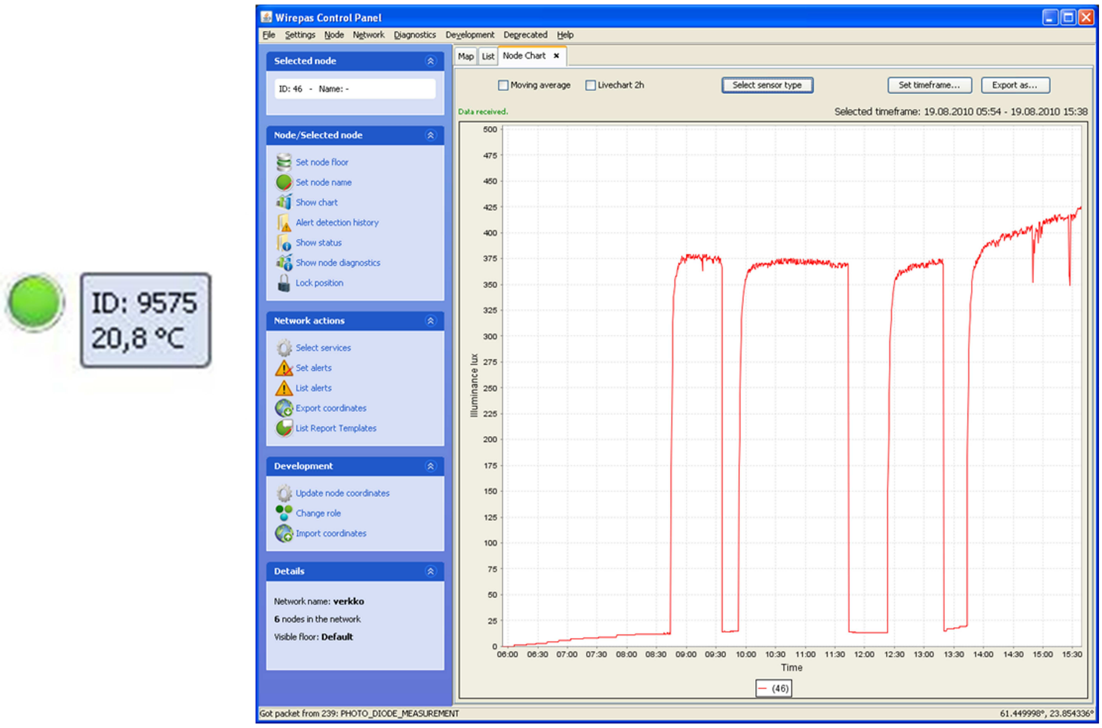

Figure 3.9: *Latest sensor value is visualized next to the node icon. Measurement graph on the right.*

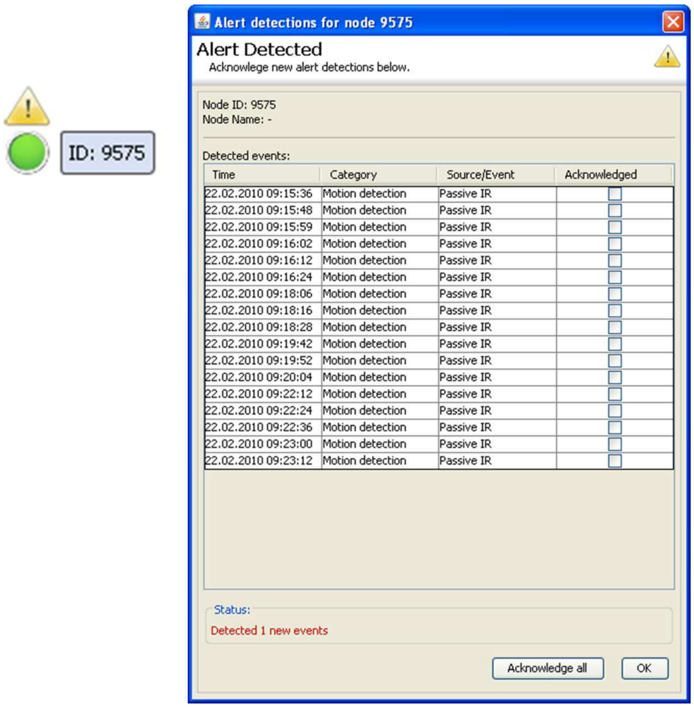

Figure 3.10: *WSN Control Panel Alert event: Triangle blinks over the node which expands to an Alert Detected dialog.*

4. SENSOR INTEGRATION PROCESS IN TUTWSN

The requirement for the sensor can come from the end-user who has a new measurement need. Usually new measurement needs are challenging. The challenge comes from choosing the right sensor from multiple potential sensors. Other possibility is that someone has already found a new sensor which is to be integrated to the WSN.

The new sensor integration to TUTWSN requires modifications to four components of TUTWSN which are presented in Table 4.1. The components are presented in the same order as the data flow from the new sensor measurement to the WSN Control Panel. The components needed to be modified are: node itself to gather measurements from the new sensor, Network Gateway (NG) to support new measurement packets and forward measurement data to a database (DB), new tables to the DB which store forwarded measurements, and finally to WSN Control Panel to display measurement data. In this thesis the focus is on the node and the sensors.

The sensor integration process consists of five major phases which are presented in Figure 4.1. Selecting the sensor is critical because everything else can be redone except this step. Reselecting the sensor consumes resources and could void rest of the steps. After selection, it takes some delivery time to receive the sensor.

The next step is hardware integration testing of the sensor. After that is integration to the node which consists of defining the radio packet, additions to the node software and measuring the power consumption of the sensor. Integration to SNAP system can be done in parallel to integration as soon as the radio packet is defined. The final step is end-to-end testing which lasts until the delivery of the WSN. The product of this process is new WSN with desired sensing abilities.

Table 4.1: *TUTWSN components needed to be modified to get support for the new sensor.*

Component	Modification or addition
TUTWSN Node	Sensor driver, application, application data structure
Network Gateway	Support for created application data structure
Database	Tables to store application data
WSN Control Panel	Measurement visualization

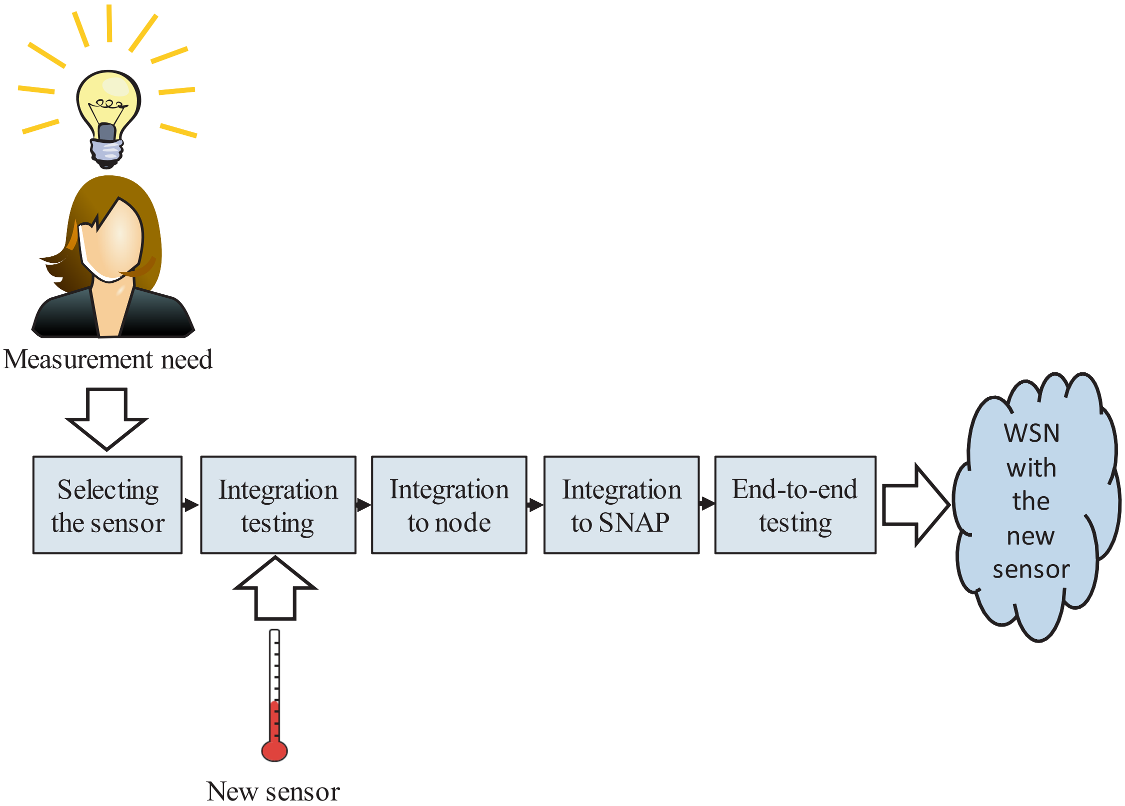

Figure 4.1: *Sensor integration steps in TUTWSN.*

4.1 Selecting the appropriate sensor

A sensor must meet the strict energy consumption requirements to be appropriate sensor for wireless use but in some cases low energy sensors are not available. Possibility of using mains power or bigger batteries must be taken in to consideration if the energy consumption of the sensor dramatically decreases node lifetime and thus the lifetime of the WSN. In this thesis the maximum reduction in the lifetime of the node caused by sensor is set to 25 % to keep the node lifetimes uniform in the network. The reduction can be compensated with bigger batteries. Using mains power is in contradiction with the idea of WSN but sometimes it is the only possibility. Using mains power also limits the locations of nodes in the network. It is also possible to separate energy supply of the sensor and the node by providing individual power supply to each. This separation enables the node to outlive the sensor and inform through WSN when sensor has depleted its power source.

A node enclosure sets limits to sensor size. If the sensor is located outside the enclosure it may need adapter printed circuit board (PCB) and its size is limited by the enclosure.

The enclosure of the TUTWSN node comprises of two parts, as seen in Figure 4.2. The node part holds the node board, batteries and the extension part which is

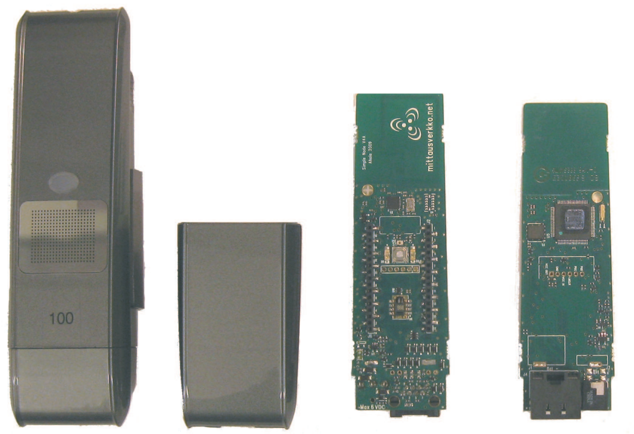

Figure 4.2: *From left to right: TUTWSN node enclosure, TUTWSN node enclosure extension part, TUTWSN node from top and bottom side.*

used for external sensors if needed. This extension is added to the end of the node part and sets limits to sensor size or circuit board of the sensor.

In the node board there should be a connector where to connect external sensors. In the simplest solution, drilled holes with headers should be enough, as in the TUTWSN node in the Figure 4.2 but this increases hand labor in the assembly process. A connector with eight pins at the end of node PCB is used in the TUTWSN nodes which helps connecting sensors. This connector contains one interrupt pin, I^2C, interrupt, and SPI buses and analog input. With enough pins it is even possible to implement future buses.

4.2 Hardware integration testing

After selecting the appropriate sensor, the integration testing processes continue. The sensor and possible adapter board is connected to the node board with wires and a small program with sensor driver is written to generate control signals and to read the sensor output.

The implementation of a typical new sensor node test program is presented in Listing 4.1. On line 2 in the Listing 4.1, the sensor output value is read from the sensor driver. The output value is printed with UART port of the node to a computer screen on the line 5. Large enough change in the quantity of the sensor is inflicted and a change in the reading of the sensor is then expected. The sensor is malfunctioning or a hardware bug is present if the change in sensor reading is not perceived with this method.

Other possibility to do the harware integration testing is to use an application which is designed to carry raw measurement data. In TUTWSN, application data

```
while(1)
{
    sensorvalue = readSensorValue();

    DebugPrint_printUint(sensorvalue);

    delayMs16(2000);
}
```

Listing 4.1: *New sensor test program on the TUTWSN node in main program.*

structure for two byte value has been used to carry measurement data of one measurement.

After this step the sensor has been evaluated and proved possible to integrate to node, it is time to consider radio packet payload. Also, measurement quantity must be defined in this step, this quantity is then visualized in WSN Control Panel.

4.3 Application Data structure design

Two different types of measurement packets exist in TUTWSN: pure measurement Application data structure and alert Application data structure. One of them is selected depending on the sensor functionality.

A Measurement Application Data structure consists of measurement data, of which size is limited by the application layer. So far, one, two or four bytes of measurement data is supported.

The Alert Application Data structure has one bit for each alert device and intuitively one is alerting and zero is not. Alert packet also has two previous alert statuses in its payload to increase reliability in case of packets being lost.

Previously a new packet type has been used for each new sensor, but because almost all of the measurement packets are identical with two byte measurement payload, a different approach is used. This different approach forms multipurpose measurement packets in which the packet payload structure is the same one, two or four bytes but new flag field is added that marks which measurement data this packet contains. This speeds up the next steps and eases testing.

Modifications to NG, DB and WSN Control Panel can be made in parallel after this step because all the needed modifications to them is now known.

4.4 Additions to the node software

Every new sensor needs additions or modifications to various node software levels. These levels are used to increase software abstraction. The levels that need to be modified or added are presented in figure 4.3. Every new type of sensor needs a new sensor application, new sensor specific driver and additions to platform specific

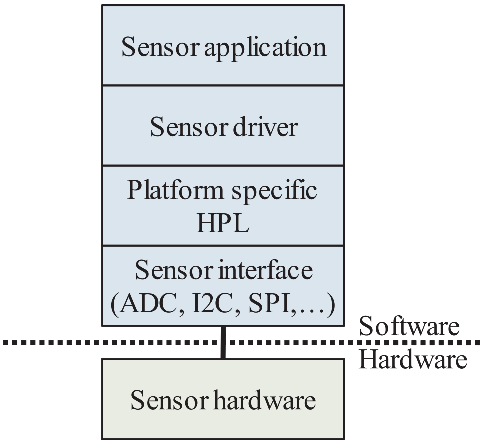

Figure 4.3: *Node sensor software levels.*

hardware presentation layer (HPL). Sensors that measure same quantities can use the same application and only use different lower level implementation.

Figure 4.4 models the required node software components in Unified Modeling Language (UML). Every application needs an application ID which is used to identify the application and index the application in application scheduler. The ID is also used in Application Data structure. The application ID is stored in app_types. Every application must contain initialization, timer and receive functions which are pointered through app_table to app_sched which handles the scheduling of the application. Initialization function is executed when node boots to initialize application specific variables and initialize the sensor driver. Timer function of application is executed every interest interval. Measurement and Application Data structure formation is performed in timer function, packet is then forwarded to lower layers in TUTWSN stack. Receive function of application enables the application to receive application specific commands or calibration values through the WSN.

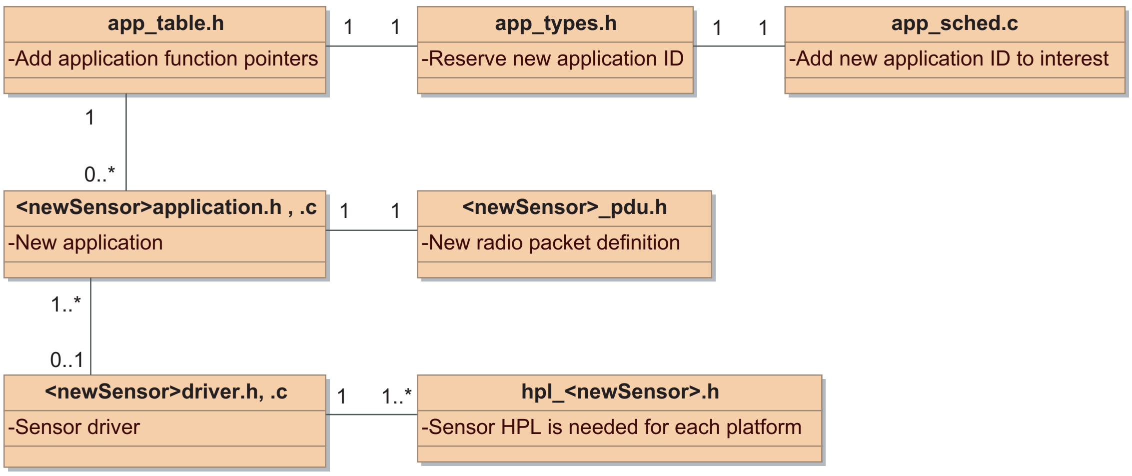

Figure 4.4: *Node sensor software classes.*

Sensor driver contains sensor hardware initialization function, get function to read sensor value and set functions for sensors which have different modes of operation. The sensor driver uses interfaces provided by the HPL to build useful abstractions and hides hardware-dependent interfaces from the sensor application. Sensor driver provides possibilities for fine filtering within a few milliseconds.

Platform specific HPL contains sensor port mappings which may vary between different platforms. HPL is the lowest level and hides hardware-dependent code from the sensor driver and other upper components and levels.

4.5 Integration to TUTWSN server infrastructure

Integration to TUTWSN server infrastructure is done in parallel to node integration which reduces the integration time.

4.5.1 Network Gateway integration

Figure 4.5 presents all the classes which have to be modified to get a new measurement to DB and WSN Control Panel.

WSN data is sent to NG and gateway node plugin, XportPlugin.java receives the TCP/IP packets and hands them to WsnMessageFactory.java. The WsnMessageFactory.java generates new measurement objects according to rules which are programmed to it. To generate new measurements, WsnMessageFactory has a WsnMessages library of classes from which is chooses the right object to be generated.

The same application ID is used in NG to form objects from Application Data structures as in the application of the node. These IDs are stored in PduTypes.java. If multipurpose measurement packets are used, NG needs to be changed only in the measurement class which separates these packets according to their flag field. With completely new Application Data structure, the changes are greater including data parsing, creating new class to WsnMessages and reserving similar application ID in NG as the application ID.

NG utilizes CPConnectorThread.java to forward new measurement objects to WSN Control Panel.

4.5.2 Database integration

New tables must be added to DB to correctly store new measurement data. The new tables to the DB are created with a script file. This script file must be executed to all DBs which are needed to support the new measurement data.

The NG stores the data to database using TutDiagnosticsDbController.java which is presented in the Figure 4.5. New database put command sentences must be programmed to this DB related class in order to store new measurement objects.

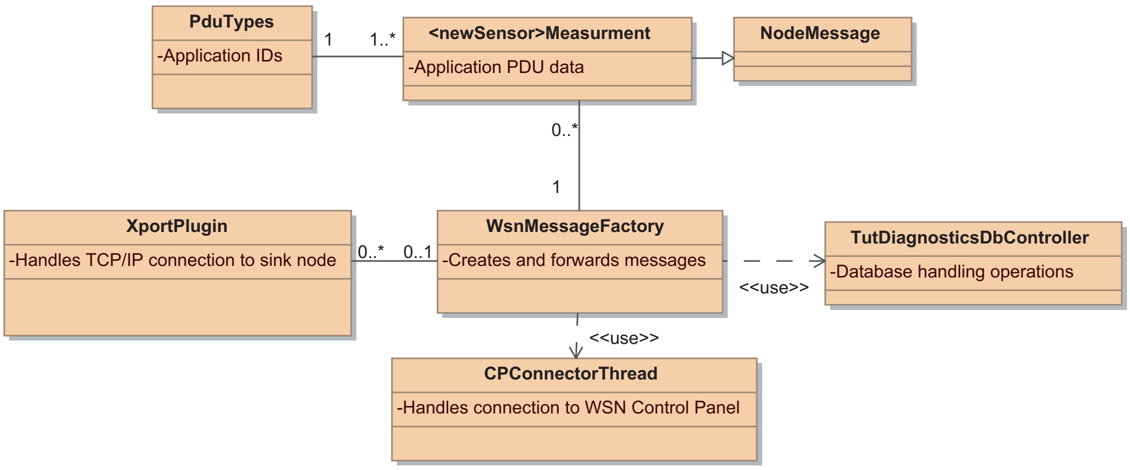

Figure 4.5: *Sensor integration related classes of Network Gateway.*

4.5.3 WSN Control Panel integration

Figure 4.6 presents classes involved in sensor measurement visualization in WSN Control Panel. The integration starts by adding newSensorSensor model class into fi.sensor.model.sensors package. A new instance of the created model class has to be added Sensors. Fi.tut.wsn.message.config.Graph must have an enumeration according to the added sensor.

Fi.sensor.ui.NodeIcon class is responsible for visualizing the recent measurement value next to the node icon. This class needs a new label according to the new sensor and the new sensor has to be added to showNewMeasurement of the NodeIcon class. The NodeIcon also needs an update method to update the visualized measurement value.

Fi.sensor.ui.SelectMeasurementDialog shows all of the available measurements in check box form, checking a box next to the measurement type displays these types next to the node icon. New sensor types have to be added to SelectMeasurementDialog check box.

Fi.sensor.model.SensorTypes class contains the quantity of the measurements and new quantities must be added to this class.

Fi.sensor.model.Options and fi.sensor.model.UISettings are responsible for storing and displaying settings of the WSN Control Panel. A new member variable has to be added to the Options class. In addition get and set functions for this new variable need to be added to the Options class. Fi.sensor.model.UISettings class needs to have a property variable to get stored settings from the DB.

SensorDataChart and fi.tut.wsn.message.config.Graph are responsible for graph form measurement visualization.

Fi.sensor.datahandler.MessageToDb.java and fi.sensor.db.DBController are re-

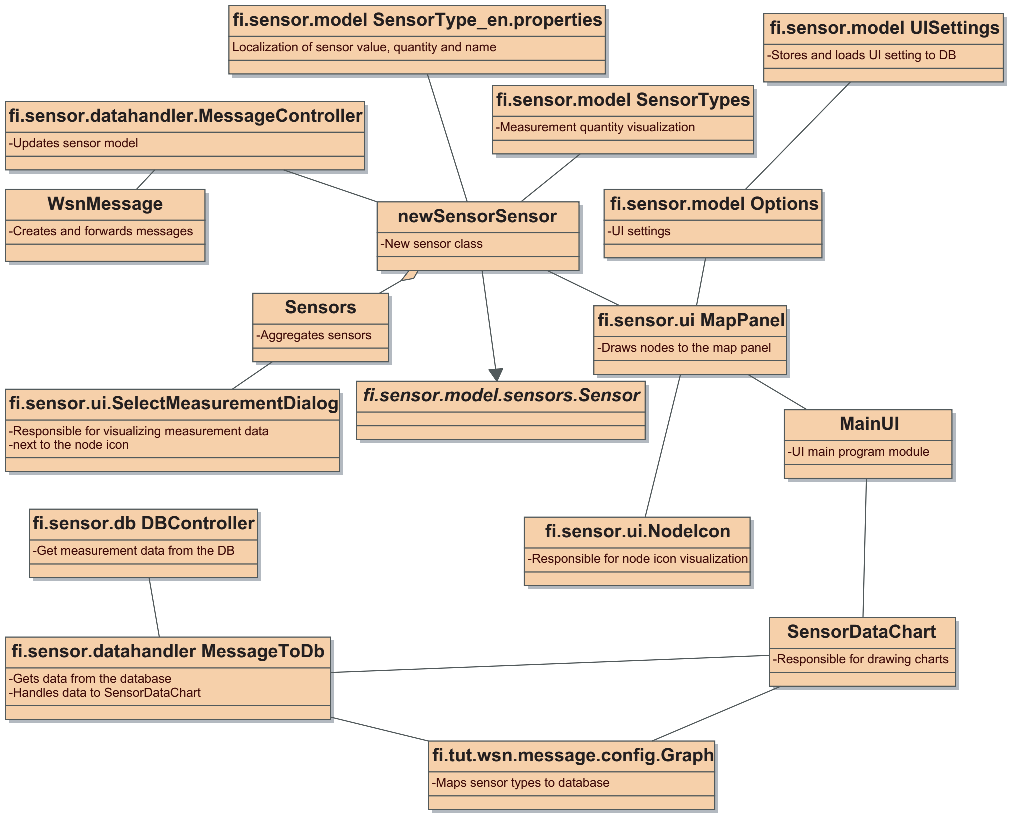

Figure 4.6: *Sensor integration related classes of WSN Control Panel.*

sponsible for providing DB access to the classes which are responsible for graph visualization.

4.6 Energy consumption testing

Energy consumption of the sensor must be measured especially if the sensor does not need mains power. This is done to evaluate the reduction in the lifetime of the node. Energy consumption of node is quite difficult to measure because activity of the node varies between deep sleep mode and high activity mode with constant radio activity. Current drawn by the node between these modes is from microamperes to tens of milliamperes which makes it impossible for a multimeter to measure.

During the years, an idea to measure energy consumption with capacitors has developed. A capacitor is used in this method as a power source and this process is presented in Figure 4.7.First, the capacitor is tuned by charging and discharging the capacitor three times. After this the capacitance of the capacitor is defined using a resistor as a current sink. Voltage and current of the capacitor are then measured every 30 seconds and from these values the capacitance is integrated. The next step is to define self-discharge rate which is inflicted by leakage current, caused by imperfect insulation materials used in capacitors, and the value is then reduced from

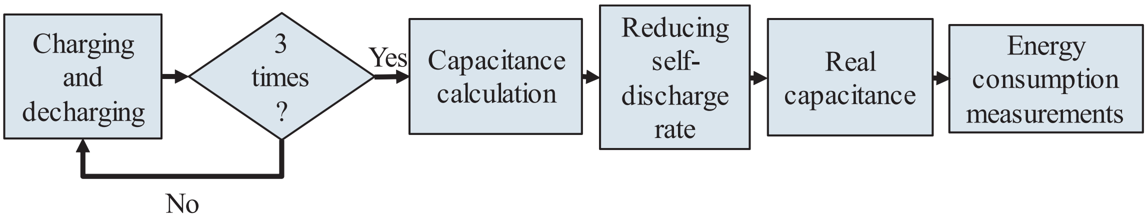

Figure 4.7: *Energy measurement preparation flow chart.*

capacitance measurements to get the real capacitance. After this, the capacitor is ready for energy consumption measurements.

The capacitor and the node with sensor are connected to DC power supply and when the node has joined the network, DC power supply is cut so that only power source of the node is the capacitor. The voltage of the capacitor is measured by the node every 30 second and the measurements values are sent via WSN. Measured voltage values provide a descending line which is visualized by WSN Control Panel and the voltage data points are exported to .CSV file format. The current and power consumption values are calculated from the .CSV file using the known capacitance. This step is redone without the sensor, and values are reduced to get the energy consumption of the sensor.

Cyclic use of sensors is mandatory. With low power sensors, the most significant factor of energy consumption of sensor is how long the sensor can be kept in the least power consumption mode until it must be turned on to perform measurements.

4.7 End-to-end testing

The end-to-end testing is done to test that every step is done and to test the sensor in real operating conditions. The WSNs are easy to deploy which makes it possible to install a network with the new sensor to an environment where end-user would install the network.

Figure 4.8 presents all the components related to the new sensor end-to-end testing in TUTWSN. First, the new sensor is connected to a node and the node is programmed with the new sensor application and radio settings according to the test WSN. Next, the node should send new sensor quantity all the way to the WSN Control Panel which is verified by visualizing the new quantity and newest measurement value next to the node icon. Historical values are also visualized, to test that values in the measurement data base are stored and fetched correctly.

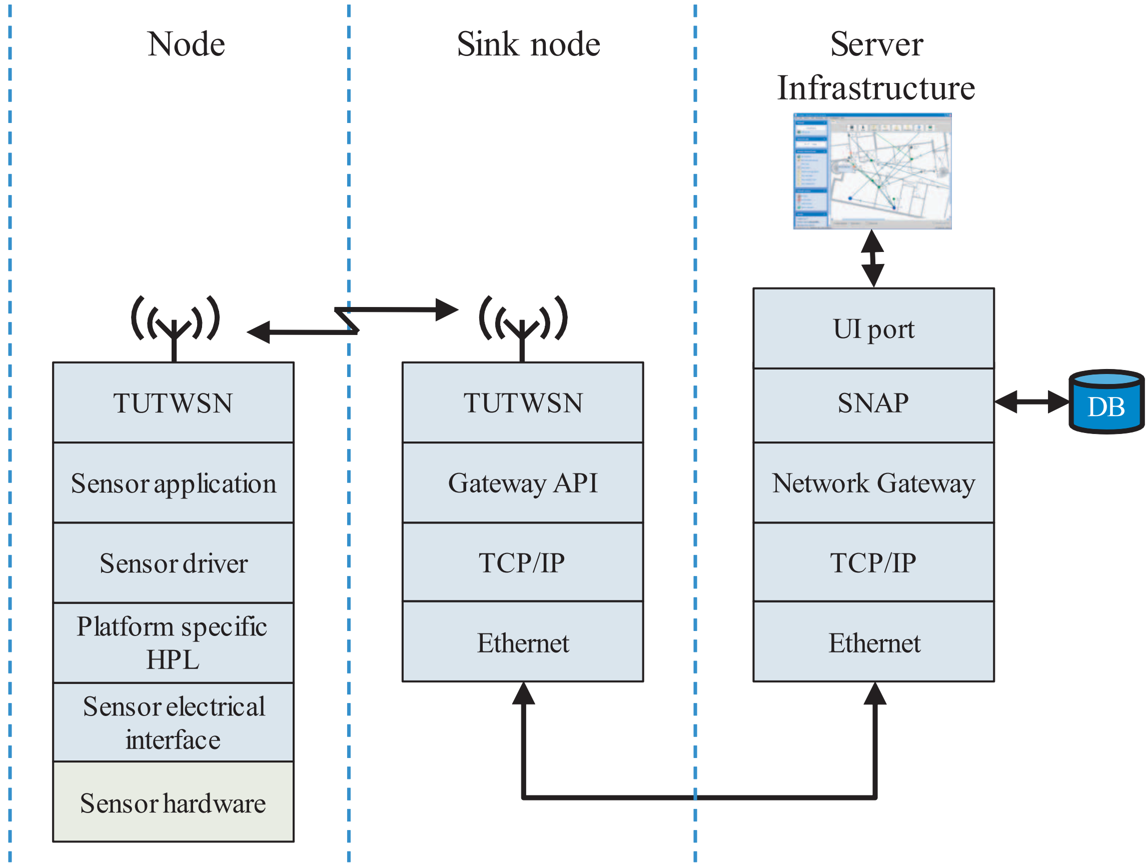

Figure 4.8: *Sensor end-to-end testing in TUTWSN.*

5. SENSOR INTEGRATION

To evaluate the sensor integration process, five different sensors are integrated. This chapter covers integration of five sensors which are integrated. The sensors are a Multipurpose Signal Sensor, an Air Velocity Sensor, a Power Meter Sensor, a Radon Sensor and a Piezoelectric Motion Detector. The sensors are integrated to test and time the sensor integration process.

5.1 Multipurpose Signal Sensor

Analog signals still exist in the field, and the majority of measurement instruments provide analog current loops from 4 to 20 mA and voltage output from 0 to 10 V. Also some measurement instruments, like resistive thermal devices (RTD), provide the possibility to measure resistance of instrument and through it a physical quantity.

There are no sensors for all these three quantities so a new sensor board PCB with ADC and digital-to-analog converter (DAC) needed to be designed. ICs for ADC and DAC were chosen: 16-bit ADC ADS1115 [81] and 12-bit DAC AD5321 [5]. These integrated circuits were chosen because they can be connected to I^2C interface. ADS1115 has four AD-channels so only one IC component is needed to measure the quantities. The block diagram of the sensor board is presented in Figure 5.1

Figure 5.2 presents measurement solutions used in the sensor board. First, the resistance measurement mode is presented on the left of the Figure 5.2. Resistance measurement requires voltage division between known voltage, known resistance and unknown resistance but to gain better accuracy, a Wheatstone bridge is used.

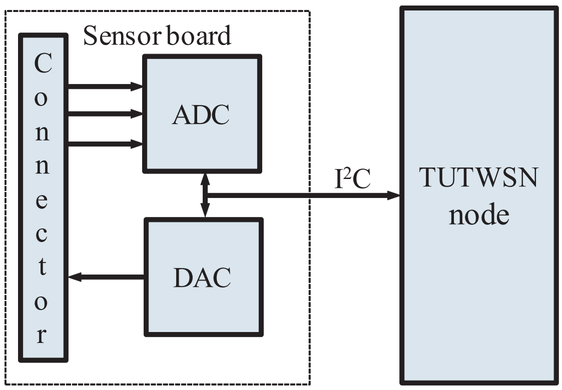

Figure 5.1: *The block diagram of the Multipurpose Signal Sensor.*

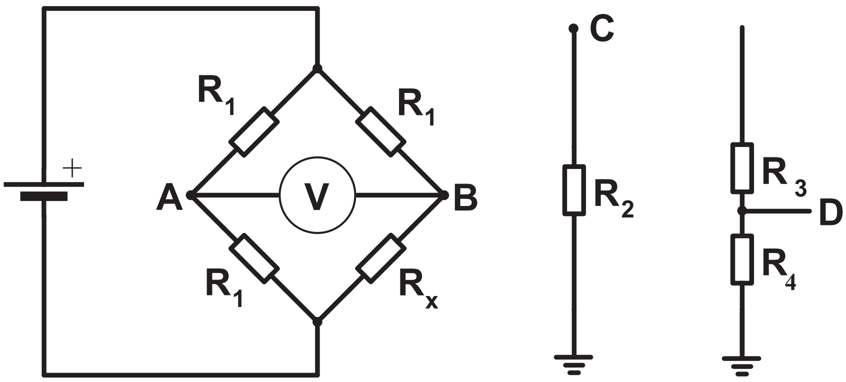

Figure 5.2: *From left to right: Wheatstone bridge with measurement points A and B, current measurement over resistor from measurement point C, voltage measurement from voltage divider from measurement point D.*

The downside in the Wheatstone bridge is that it requires two measurement points which are marked with A and B. Second, the current measurement mode. A simplest way to sense and measure the current is a small precision resistor and voltage over it is measured. This is visualized in the middle of the Figure 5.2. Voltage is measured from point C. Third, the voltage measurement mode. Voltages over the operating voltage of the node are impossible to measure without some kind of proportioning, so to measure voltages up to ten volts a resistor division is used. This is visualized on the right in the Figure 5.2.

The integration testing step includes the testing of the ADC and its three measurement modes: resistance, current and voltage and the DAC. A small test program with functioning sensor driver for ADS1115 and AD5321 is written and it is used to get correct calibration values for quantities. Figure 5.3 presents testing of the resistance measurement mode.

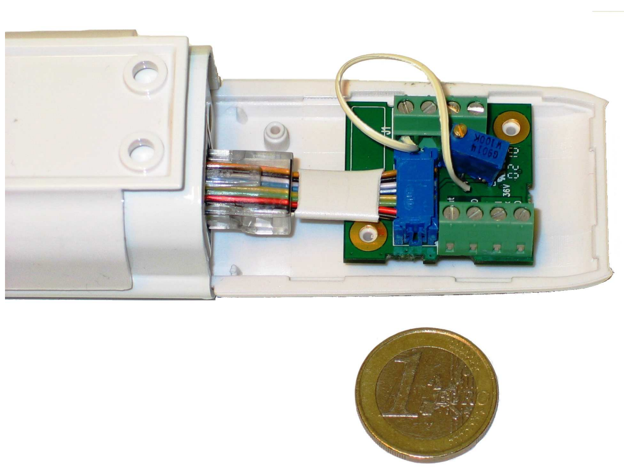

Figure 5.3: *The Multipurpose Signal Sensor with potentiometer used for testing the resistance measurement mode.*

The voltage measurement mode is tested by connecting adjustable voltage source with voltage meter to the voltage input of the sensor board. The ADS1115 measures voltage from point D. These values are logged and these provide the Figure 5.4. The voltage response of the sensor is linear in the voltage measurement mode and the voltage obeys the equation

$$V_{IN} = 0.0004 \times ADC_{value} \tag{5.1}$$

Figure 5.5 presents the relation between the ideal voltage and the voltage calculated with the equation 5.1. The dotted line is the ideal line which does not contain error and the solid line is the calculated line which differs from the ideal line. From Figure 5.5 it can be observed that the calculated values are smaller than the ideal values. The maximum difference between the ideal and the calculated line is 5 % close to origin but the difference drops to only 2 % after 2 volts. In the ideal situation, without rounding error, the accuracy of the voltage measurement mode is previously gained 5 % and this will be thoroughly tested in the end-to-end testing step.

The current measurement mode is tested by applying current to the current input of the sensor boards and measuring the current with current meter. Generated voltage over a resistor is measured from the point C and the values of ADS1115 are logged. Logged value and current meter reading pairs are presented in Figure 5.6.

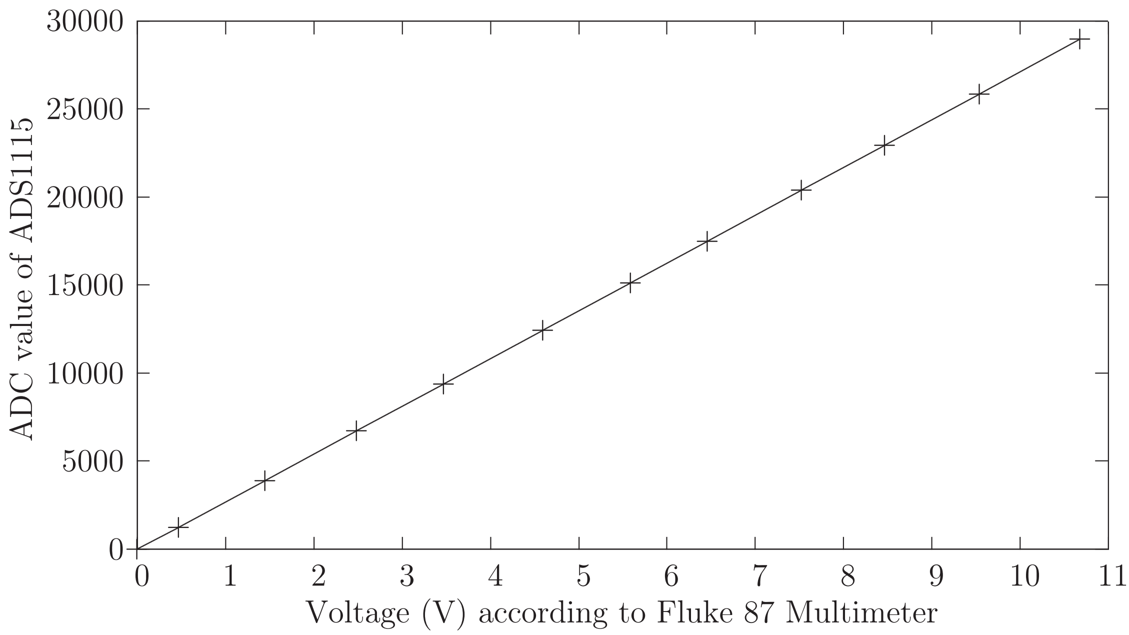

Figure 5.4: *ADC value of ADS1115 with different voltages*

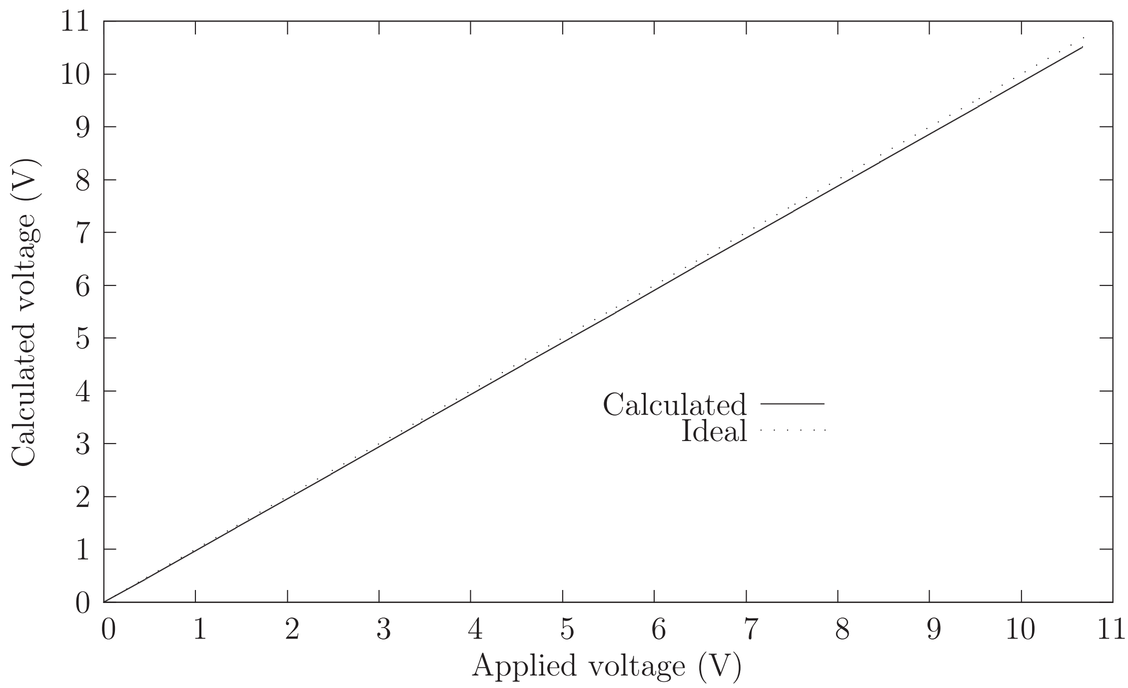

Figure 5.5: *Applied voltage and the result voltage.*

The current response of the sensor is also linear in the current measurement mode and the current can be calculated from the ADC values using the equation

$$I_{IN} = 0.0314 \times ADC_{value} \tag{5.2}$$

The ideal current and current calculated with the equation 5.2 are presented in Figure 5.7. The dotted line is the ideal line which does not contain error and the solid line is the calculated line which differs slightly from the ideal line. The maximum difference between the ideal and the calculated line is less than 1 %, 0.57 % at around 4 mA. The ideal accuracy for the current sensing is 0.57 %.

The third measurement is the resistance measurement mode. ADS1115 measures voltage between points A and B, as presented on the left in the Figure 5.2. To test this measurement mode, a potentiometer is connected to the resistance input of the sensor board, this is presented in the Figure 5.3, and resistance is measured using Fluke 87 Multimeter. Gained potentiometer resistance value and ADC value of ADS1115 pairs are presented in Figure 5.8. The curve in the Figure 5.8 provides resistance –ADC value relation accurately with third order equation

$$R_{IN} = -2 \times 10^{-13} \times ADC_{value}^{3} + 5 \times 10^{-8} \times ADC_{value}^{2} + 0.0269 \times ADC_{value} \tag{5.3}$$

The implementation of the curve would be extremely resource consuming to be

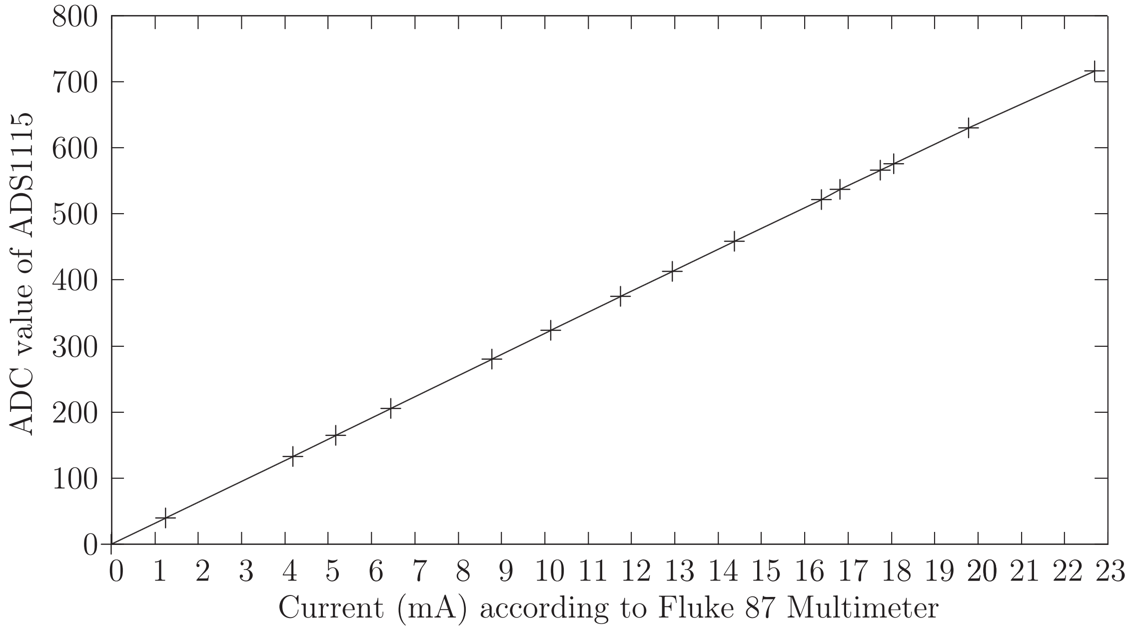

Figure 5.6: *ADC value of ADS1115 with different currents*

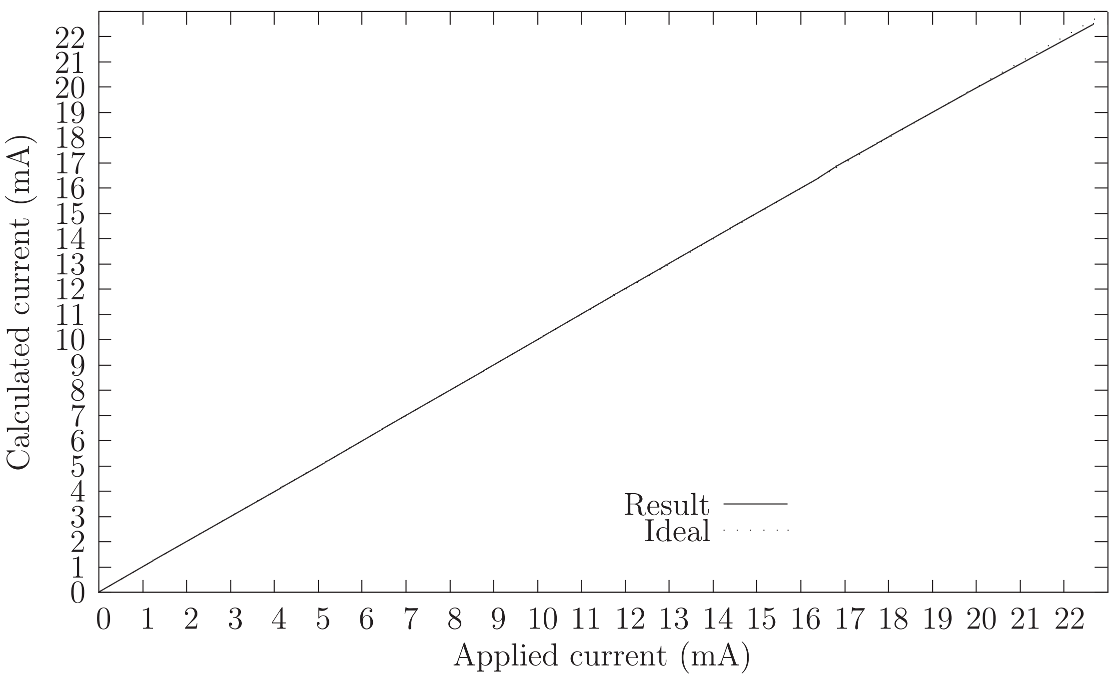

Figure 5.7: *Applied current and the result current.*

implemented on the node. To ease the resistance calculation, the curve is cut into seven linear parts, presented in Figure 5.9. The downside of this is that fitted curves increase the error but are faster to calculate and less resource consuming.

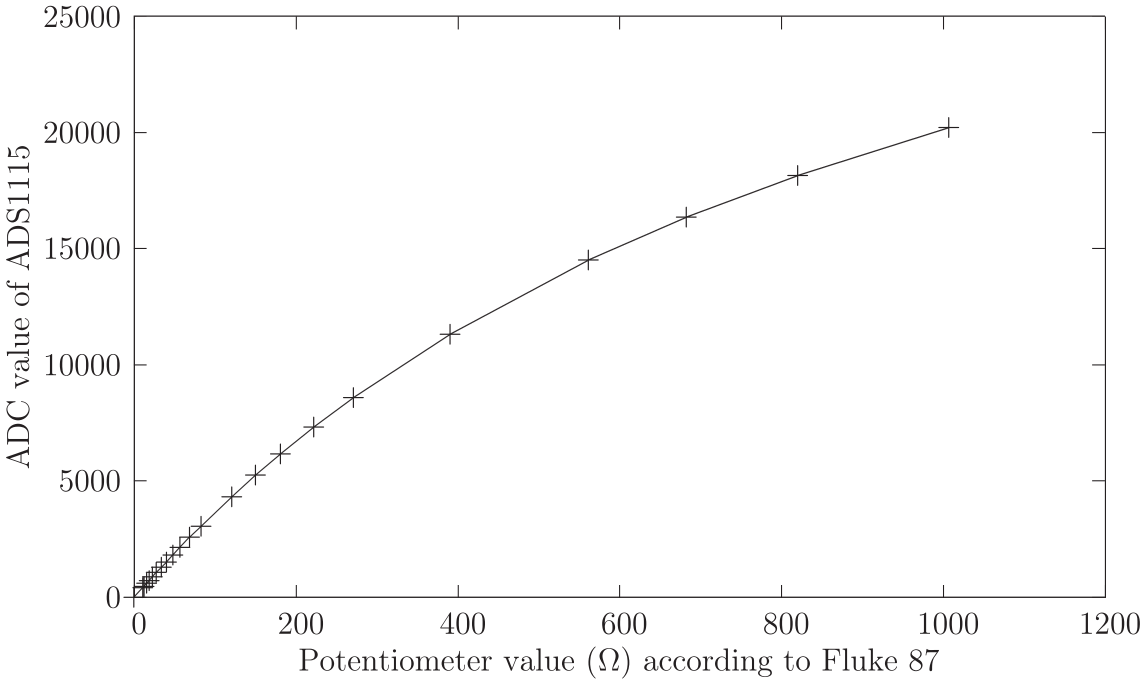

Figure 5.8: *ADC value of ADS1115 with different potentiometer values*

Sensor board also contains acting component in the form of a DAC. The DAC is tested by setting its output and increasing it in ten unit steps. The output of the

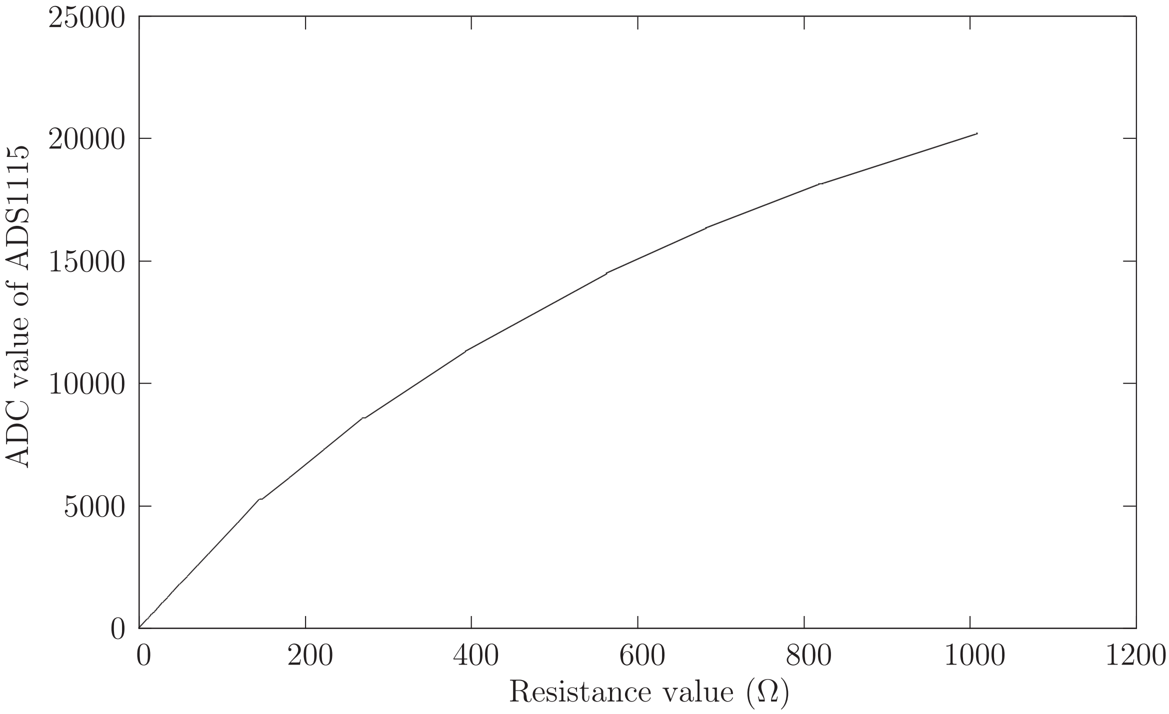

Figure 5.9: *Fitted resistance lines and resistance.*

DAC is measured with Fluke 87 Multimeter from the output of the sensor board. These value pairs provide Figure 5.10. The data sheet of DAC AD5321 [5] provides the ideal output voltage formula as

$$V_{OUT} = \frac{V_{DD} \times D}{2^N} \tag{5.4}$$

where V_{OUT} is the output voltage generated by the DAC, V_{DD} is the supply voltage, in this case it is the voltage of the node, 2.5 volts, D is the decimal equivalent of the binary code, N is the DAC resolution which in this case is 12 bits. In the ideal situation one DAC bit equals 0.610 mV. From the equation of the slope in the Figure 5.10

$$V_{OUT} = 0.614 \times DAC_{value} \tag{5.5}$$

one DAC bit is 0.614 mV which is only 0.004 mV larger than the ideal. The ideal accuracy of the DAC is 5.5 % close to the origin but after 700 mV the accuracy is less than 1 %.

For the actuating part, the application uses TUTWSN actuator Application Data structure which is presented in Figure 5.11. The different actuators are specified with CmdID and the action which is required from the actuator is in Act_cnt.

The end-to-end testing has to be done to each measurement mode individually. First, the voltage measurement mode is tested by connecting the voltage source and

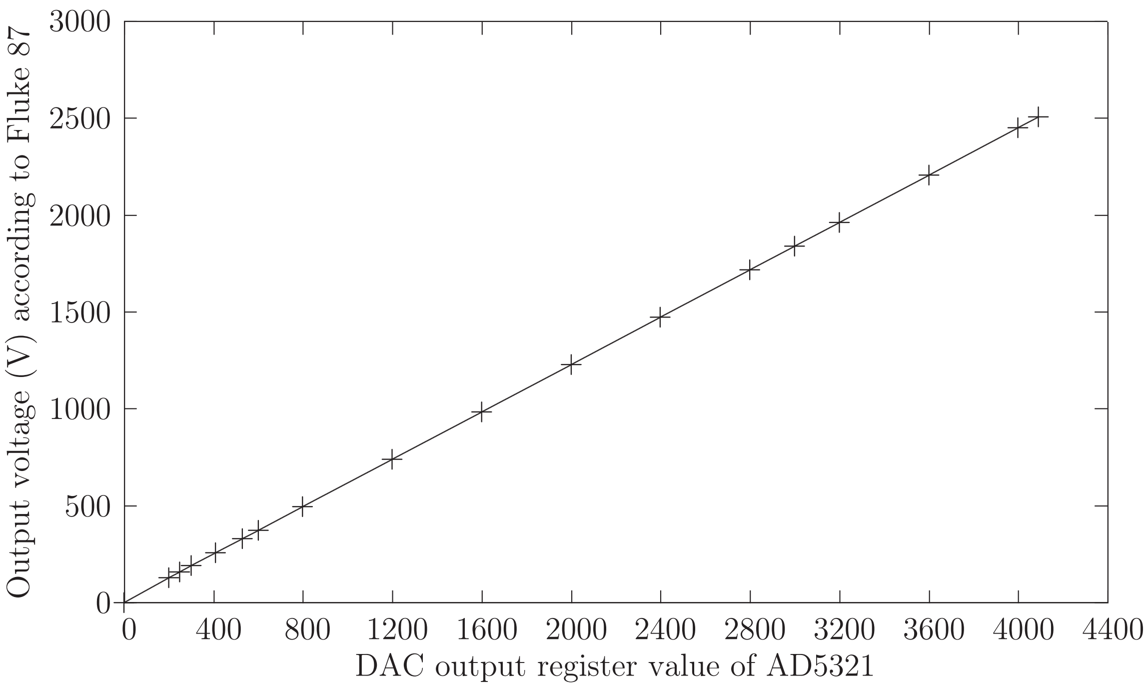

Figure 5.10: *Output value of AD5321 with different DAC output register values.*

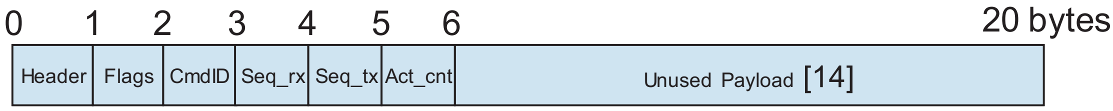

Figure 5.11: *TUTWSN actuator Application Data structure.*

voltage meter to the voltage input of the sensor board. The voltage is increased in 0.5 voltage steps and the results from WSN Control Panel are presented in Figure 5.13.

The values are also compared to Fluke Multimeter values and the percent difference graph is presented in Figure 5.14. The Fluke is not absolutely correct but it gives a good calibration and comparison point for the first version of the sensors. The difference is greatest where values are the smallest and quickly shrinks to less than -2.5 %. The error is almost constantly -2.0 % so to correct it simply requires adding 2.0 % to the measured value.

Second, the current measurement mode is tested. One problem was that the output of a current source was not stable until a bigger load was connected to it. This bigger load was a bigger resistor and the sensor current input is connected in series with this resistor. The results of the end-to-end testing is presented in Figure 5.15.

In end-to-end testing, the current is increased in steps and the calculated current value of the node is compared to the value of the Fluke Multimeter. The results are presented in Figure 5.16. The biggest difference is around 1.00 mA, where the difference to the value of Fluke Multimeter is -3.84 %. The difference drops to less than ± 0.5 % by 3.00 mA. If the Fluke Multimeter is concerned ideal the accuracy of the current measurement mode is ± 4.0 %.

Third measurement mode, the resistance measurement mode, is tested similarly. The results are presented in Figure 5.18 and the difference graph is presented in Figure 5.17. The behavior of the difference graph is caused by the fitted lines which try to estimate the resistance. The difference with 1000 Ω range is -4.2 % at the greatest close to 200 Ω and shrinks to around -1.5 % only to drop to -3.5 % around 800 Ω. The accuracy of the resistance measurement mode is ± 4.5 %.

The Multipurpose Signal Sensor is measurement device but it contains three different measurement modes. Multipurpose measurement radio packet is not suitable for the sensor because multipurpose radio packet has only space for one measurement

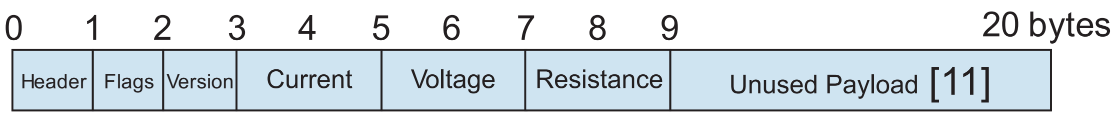

Figure 5.12: *Multipurpose Signal Sensor Application Data structure.*

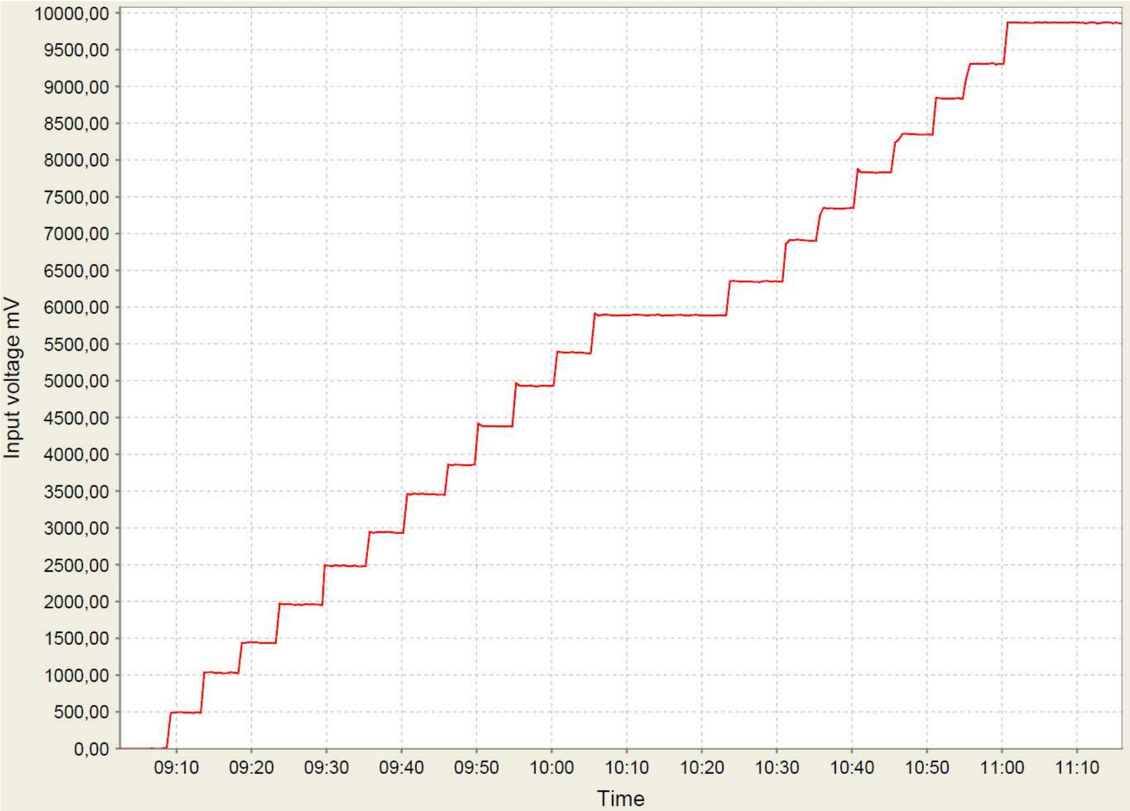

Figure 5.13: *Voltage measurement mode end-to-end testing.*

and it would increase WSN traffic to send the measurements in multiple packets. The solution to traffic problem is to design a new packet which carries three quantities: current, voltage and resistance value. Figure 5.12 presents the new Application Data structure. The payload is filled every time the application is executed. Previously there has only been one measurement in each radio packet so a new packet handling module needed to be implemented to the NG. This new module is able to manufacture multiple measurement from one packet.

The conclusion table is presented in Table 5.1. The electrical properties in the table are the worst case scenario which is the resistance measurement mode.

The future development of the Multipurpose Signal Sensor will include the integration of pulse counter which will be used to count pulses from various sources, one of the source could be power meter. The developing of the actuating part will include industrial compatible current and voltage commands.

Table 5.1: *The Multipurpose Signal Sensor conclusion table*

Memory				Electrical			
Program bytes	**Data bytes**	**Program %**	**Data %**	**Current (mA)**	**Voltage (V)**	**Power (mW)**	**Energy (μJ)**
3920	16	3	0.4	1.0	2.5	2.5	2.5

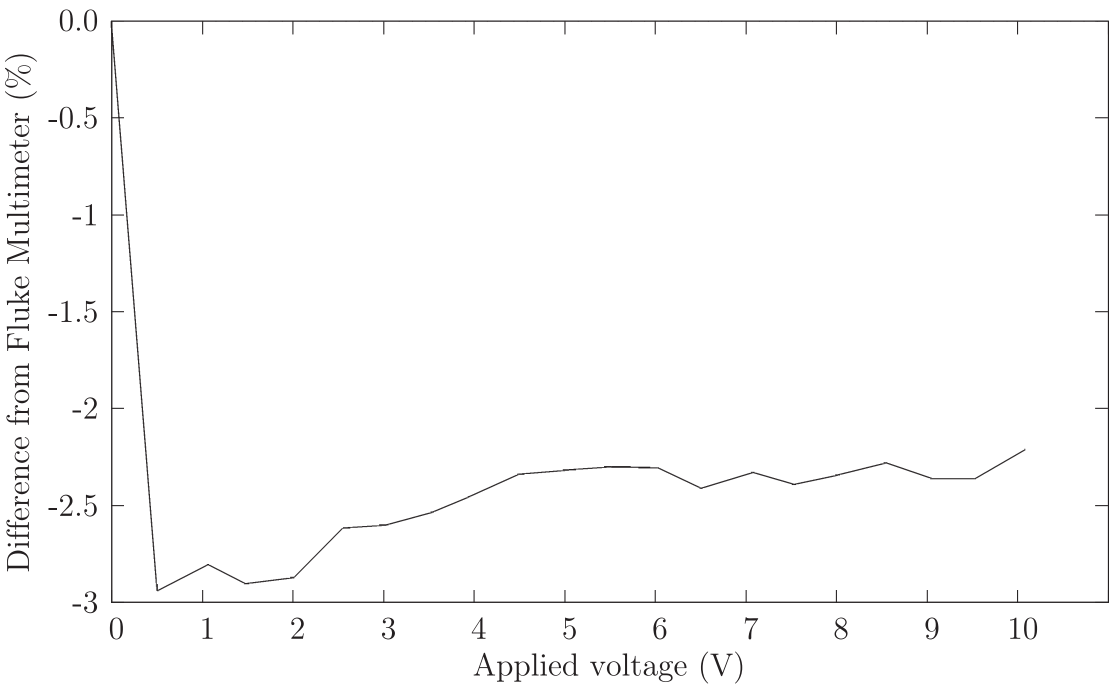

Figure 5.14: *Voltage measurement mode error.*

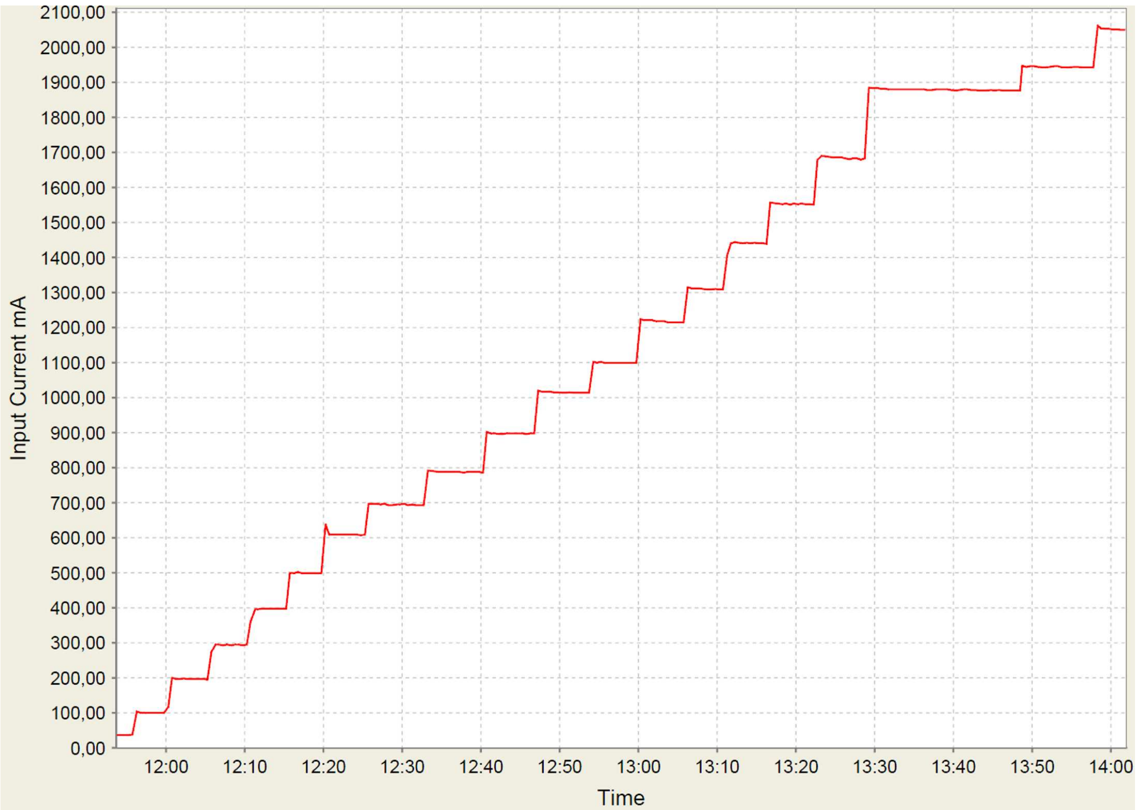

Figure 5.15: *Current measurement mode end-to-end testing.*

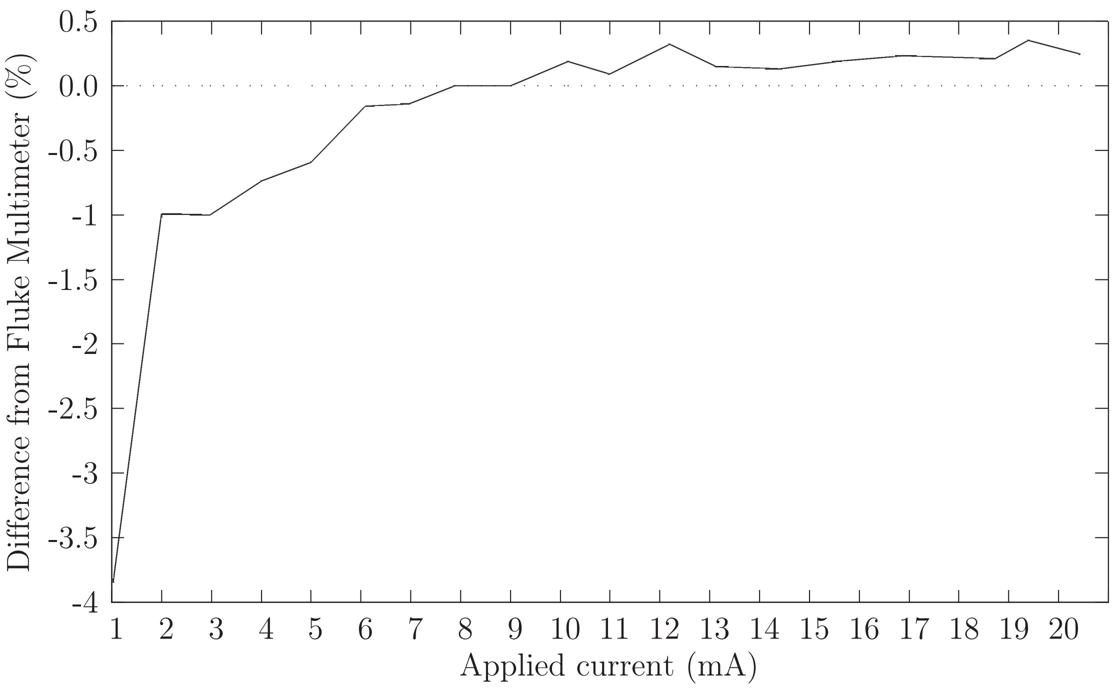

Figure 5.16: *Current measurement mode error.*

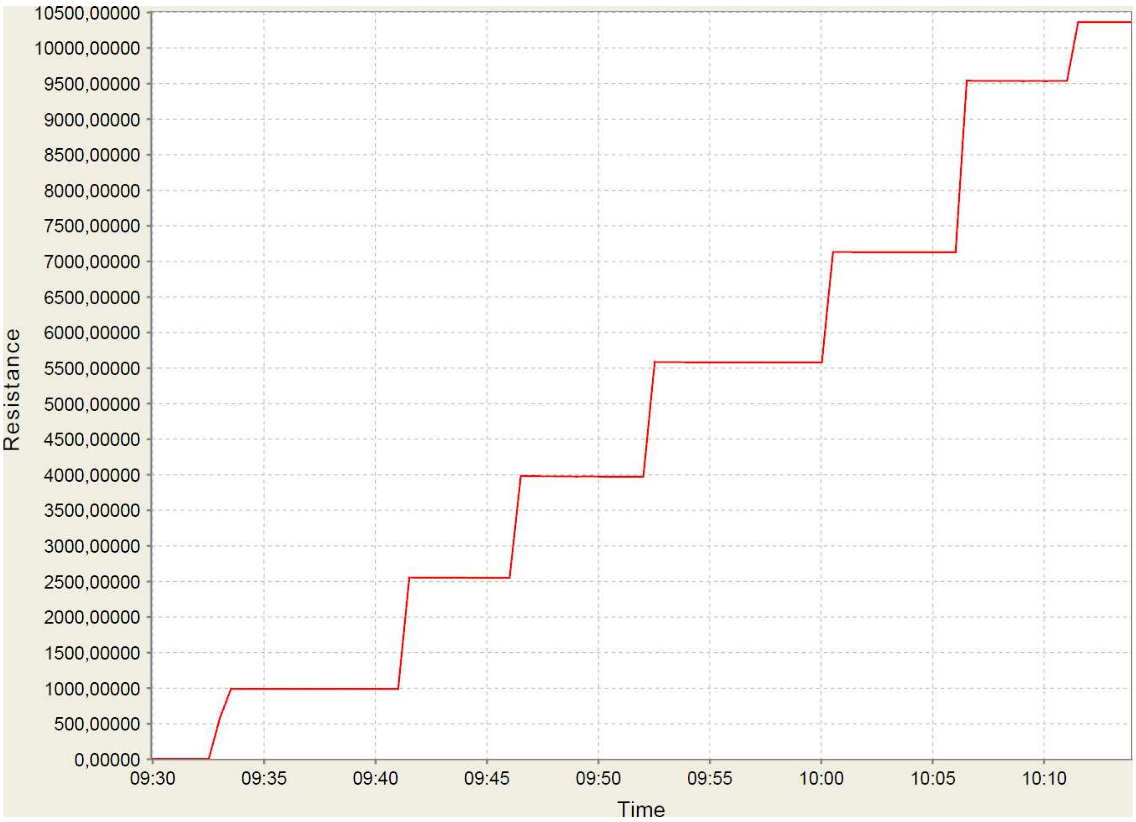

Figure 5.17: *Resistance measurement mode end-to-end testing.*

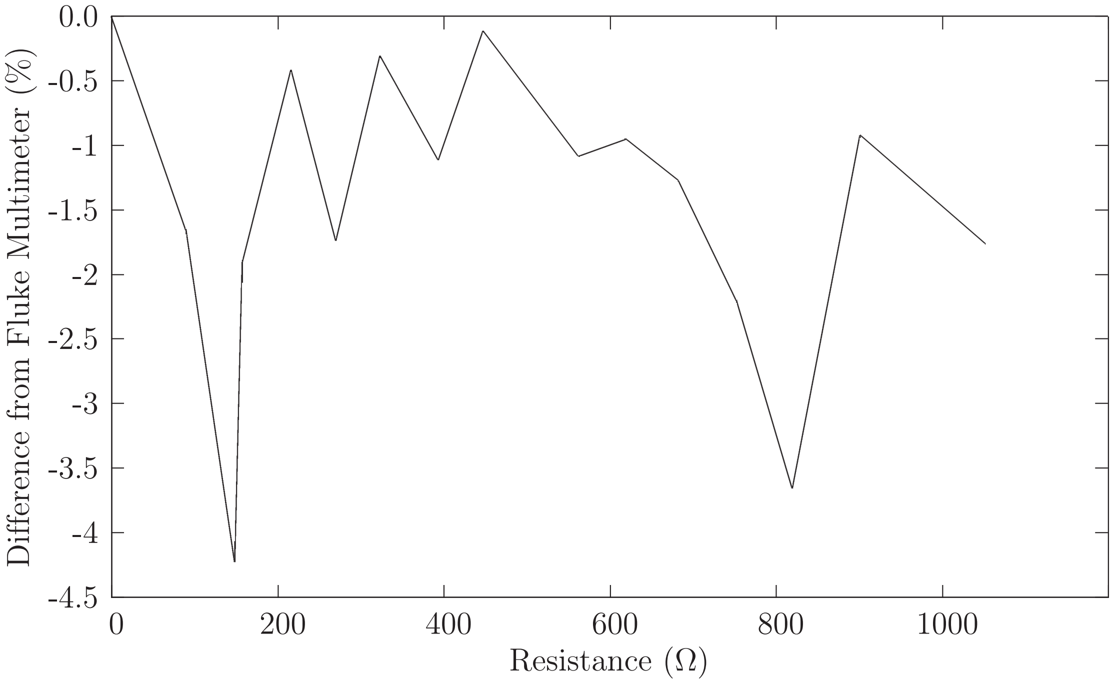

Figure 5.18: *Resistance measurement mode error.*

5.2 Air Velocity Sensor

The air velocity sensor is needed in one of the deployments where air flow sensing nodes form a significant part of the WSN. In this deployment, gathering data is only possible using WSN because the deployment location is an old building. The WSN is used to improve energy efficiency of the building by measuring possible energy inefficiencies in the structures and in the heating system. The end user who wants the WSN requires that air velocity sensor must be capable of measuring air velocities between 0.0 to 10.0 $\frac{m}{s}$.

A wide range of air velocity sensors exist in the market and the most of them have the same basic working principle: hot-film anemometer principle. In hot-film anemometer a thin metallic film is heated and the flow cools it [9]. From the temperature it is then possible to conclude the flow velocity [9].

Two air velocity sensors were considered: flow sensor SS20.260 [65] by Schmidt and EE575 [15] by E+E Elektronik, both fulfilling the air velocity requirements set by the end-user. The SS20.260 requires 24 V supply voltage and 30 mA supply current, The EE575 requires 10 V and 70 mA so both sensors require high supply current which makes it impossible to use sensors with only the power supply of the node. The SS20.260 is designated to industrial use and its outputs are 0 to 10 V or 4 to 20 mA, the SS20.260 has also build-in temperature sensor and some fault LEDs. The EE575 provides outputs of 0 to 5 V or from 0 to 10 V and provides only air velocity measurement. The EE575 was selected based on these data sheet facts because it was more suited for node electricity levels and provided only the needed air velocity. The sensor price was not one of the selecting factors but The EE575 costs only 20 % of the price of SS20.260.

The Multipurpose Signal Sensor is supposed to be used with the air velocity sensor but the Multipurpose Signal Sensor was not available in time to be used in the deployment. The lack of Multipurpose Signal Sensors force the design of a PCB which would lower 0 to 5 V to 0 to 2.5 V and provide screw connector for the power supply and for the EE575. The supply voltage of EE575 is too high for the regulator of the node so the PCB also has to have a bigger regulator. The output voltage of the EE575 is fitted to node levels with a simple resistor division. The block diagram of the designed air velocity sensor is presented in the Figure 5.19. Figure 5.20 is a picture of the EE575 connected to the Multipurpose Signal Sensor which will be used in future deployments.

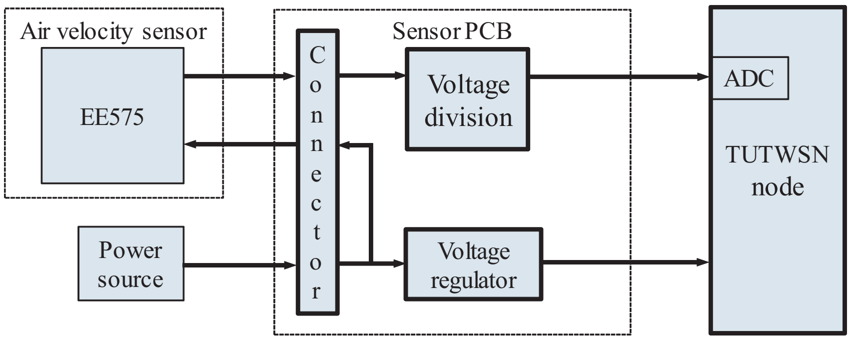

Figure 5.19: *Block diagram of the air velocity sensor.*

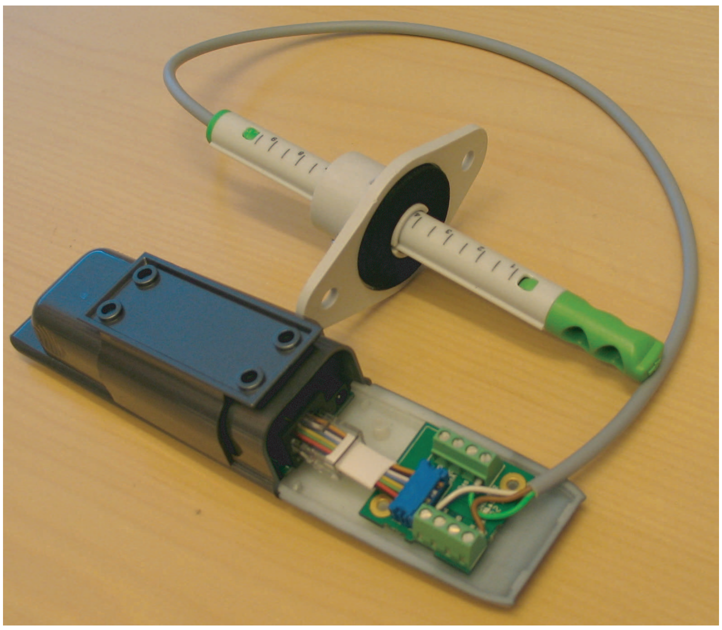

Figure 5.20: *The air velocity sensor.*

The air velocity sensor needs air flow in which it will be tested. A test bench with adjustable air flow was built from a fan, a cardboard tube and an adjustable voltage source. The air velocity test bench is presented in Figure 5.21 and a photograph of it is presented in Figure 5.22.

The data sheet of the fan states that the maximum flow was 40.78 cubic feet per minute (CFM) which was converted to $\frac{m}{s}$ in a tube with a diameter of 8 cm. 1 CFM equals 4.72×10^{-4} $\frac{m^3}{s}$ and $\pi \times r^2$ provides the area of the tube : 0.005 m^2, with these numbers, the maximum velocity is 3.85 $\frac{m}{s}$. With the test bench, it was possible to do the integration and end-to-end testing steps.

The air velocity sensor is a pure measurement device so the packet choice is easy: use pure measurement packet and use multipurpose measurement packet. This also dictates data visualization in WSN Control Panel to be measurement visualization. The Application Data structure of multipurpose measurement packet is presented in Figure 5.23

The output of the EE575 is linear in comparison to air velocity with a small base level of 200 mV. This base level was eliminated and the linearity of the output was used to calculate air velocity in sensor driver. The sensor output varied in constant air flow as much as 1.5 $\frac{m}{s}$ between normal two minute TUTWSN measurement intervals so a longer average filtering was needed and it was done with self-scheduling the sensor application. The self-scheduling sensor application calls itself ten times once a second when the TUTWSN measurement interval triggers and a longer averaging

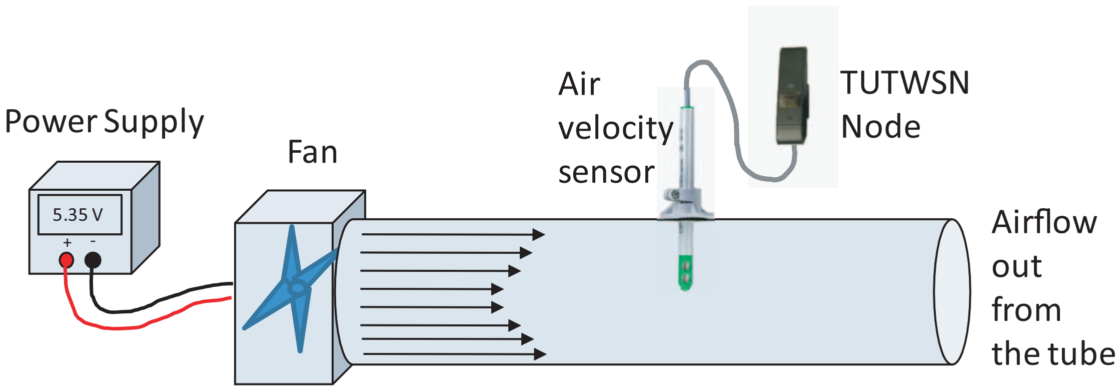

Figure 5.21: *The test bench built for the air velocity sensor.*

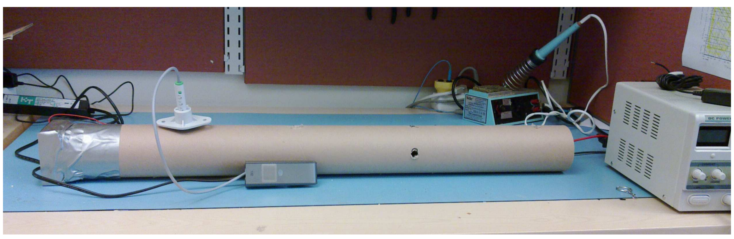

Figure 5.22: *Photograph of the test bench.*

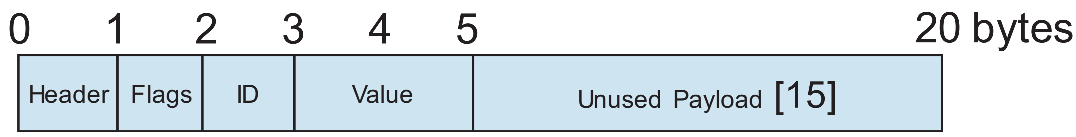

Figure 5.23: *Application Data structure used by the air velocity sensor.*

is gained.

The Table 5.2 presents the memory usage and electric characteristics of the air flow application. The electrical characteristics present the worst case scenario in the maximum air flow.

The end-to-end testing was done with the same test bench as before. The voltage of the test bench fan was increased and the results are presented in figure 5.24. In the beginning the fan voltage is 4.0 volts and was increased in one volt steps. From the measurements the air flow is increased 0.4 $\frac{m}{s}$ for each one voltage step.

After the WSN was delivered and installed by the end-user, in routine checks it was observed that the air flow values are quite noisy. The problem was investigated and as can be seen from the figure 5.25 the signal is noisy. As a solution it was evaluated that the averaging could be increased but this would decrease the response time of the sensor. Averaging can be done in WSN Control Panel which proved to be a good solution. Also, it is not excluded that the flow profile of the duct is whirling or for other reasons the flow is not constant.

The requirement for mains power is a major limitation to the placement of the sensor and a solution to this problem was needed. The supply voltage of air flow sensor enabled the use of widely available 12 volt batteries but if the sensor is constantly powered on, the battery does not last very long time, which can be observed from Table 5.3. The solution to the lifetime problem is to switch power off when the air flow sensor is not measuring. The effects of switching on energy consumption are presented in Table 5.4. A problem from switching the power ON and OFF is that the voltage of the air flow sensor needs at least 5 seconds to stabilize before the output of the sensor is stable enough. This extends the total power-on time of the sensor to 15 seconds.

The end-user would like the WSN to calculate volume flow rate which will be done in the future by sending the channel diameter from WSN Control Panel all the way to the node. Air flow sensor with a bulky 7.2 Ah battery is delivered to the

Table 5.2: *The conclusion table of the air velocity sensor*

Memory				**Electrical**			
Program bytes	**Data bytes**	**Program %**	**Data %**	**Current (mA)**	**Voltage (V)**	**Power (mW)**	**Energy (J)**
1177	12	0.9	0.3	70.0	12.0	840.0	8.4

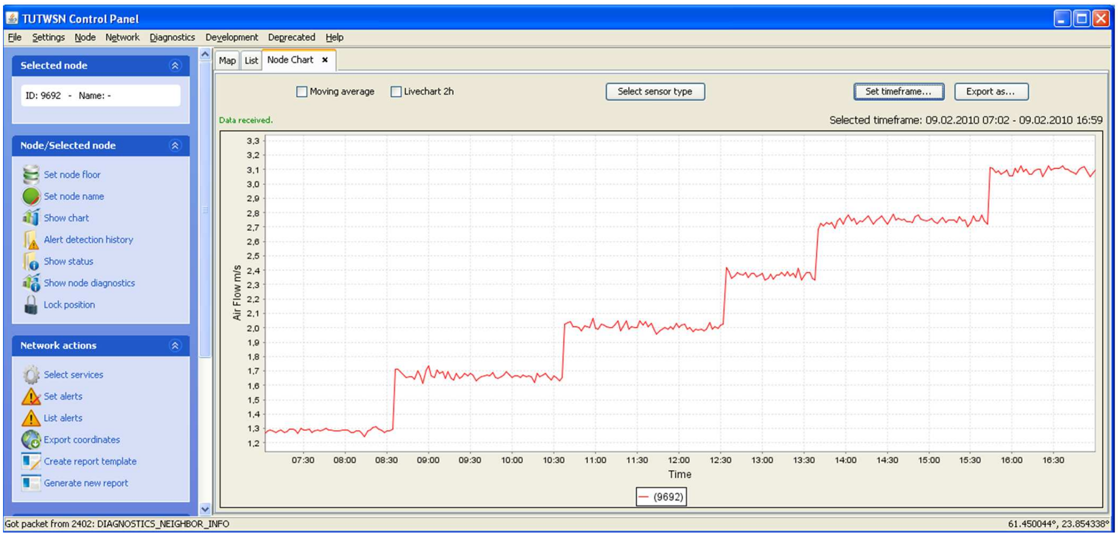

Figure 5.24: *Air flow in the test bench according to EE575 Miniature Air Velocity Transmitter.*

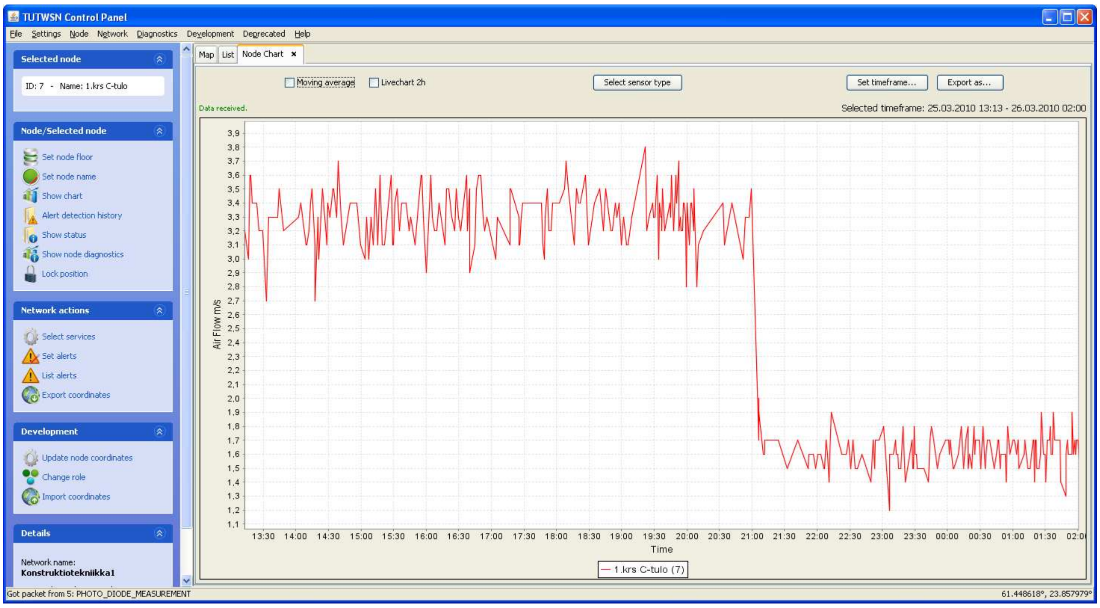

Figure 5.25: *Measurement values of air flow sensor from a real air duct.*

Table 5.3: *Lifetime of the battery powered air flow sensor in hours.*

Interest interval	Battery 12 V, 1500 mAh 64 800 J	12 V, 2200 mAh 95 040 J	12 V, 7200 mAh 311 040 J
Constantly ON	21.4	31.4	102.9
30 s	42.9	62.9	205.7
1 min	85.7	125.7	411.4
2 min	171.4	251.4	822.9
5 min	514.3	754.3	2468.6

Table 5.4: *Energy consumption of the battery powered air flow sensor.*

Interest interval	**Sleep time** (s)	**Sensing time** (s)	**Activity** %	**Energy** (J)
Constantly ON	0	10	100.0	-
30 s	15	15	50.0	6.3
1 min	45	15	25.0	3.2
2 min	105	15	12.5	1.6
5 min	345	15	4.2	0.5

end-user and further developing will be made according to the feedback.

The future development of the sensor will most likely be a differential pressure sensor which would enable AA batteries to be the main power source and thus reducing the size and weight of the current sensor. AA battery power source and smaller sensor will make the deployment of the air velocity sensor easier.

5.3 Power Meter Sensor

The most of the residences have an electric meter which measures the amount of electrical energy supplied to it. In older residences it is analog electric meter and in newer it is digital. New wave of electric meters are digital meters which communicate consumed electric energy consumption to energy company via a telecom network.

The amount of energy is usually measured in kilowatt hours which makes it almost impossible to notice the affect of one appliance to the total consumption, and the usage cost of the appliance from the increased electric bill.

Environmental awareness and demand for more energy efficient appliances and machines has opened market for smart energy metering, for an example ZigBee [4] Smart Energy [3]. These smart energy metering options enable to sub-meter electrical energy consumption within a home or apartment even from the individual wall sockets. The key point behind is that consumers will better be able to reduce energy consumption when they have easy access to consumption data [58].

The Power meter sensor is a sensor that is able to calculate power consumption of an appliance. There are two possible methods to measure power: galvanic and non-galvanic. In galvanic method, the current goes through a small resistor and a voltage over it is measured. In non-galvanic method, the power meter sensor is based on the fact that the voltage in electric grid is constant 230 V_{rms} and only the current varies depending on the power consumption. Equation

$$P = U \times I \tag{5.6}$$

shows the relation between the power P, the voltage U and the current I. The current creates alternating magnetic field around the wire and when directed to a coil the magnetic field creates voltage which is relative to current. The power is directly proportional to the voltage which is measured from the coil. The non-galvanic method is chosen because a galvanic sensor would require extensive CE testing to receive CE approval and it would require trained electrician to install a galvanic sensor.

There are power sensors available on the market but all of them have features which are not needed, like displays and buttons to reset, so the power sensor is designed from the beginning without the unneeded features. The sensor is based on a current transformer vacuum coil [85] manufactured by VACUUMSCHMELZE [84], a rectifier, a capacitor, and a zener–diode. Wire of an electric cord is pulled through the vacuum coil and a plug and a socket are put to the both ends of the cord. A block diagram of the implementation is presented in Figure 5.26 and a picture of the sensor is presented in Figure 5.27.

In the integration testing step, the output of the power meter sensor is connected to one ADC inputs of the node. The power meter sensor and STEVAL-IPE006V2 High end meter Demonstration board [72] by STMicroelectronics are connected in series and the load is varied to get the output of power meter sensor related to values of STEVAL-IPE006V2. In this method the STEVAL-IPE006V2 values are considered to be absolute without any error which is enough for the WSN applications. The relation is presented in Figure 5.28.

The Figure 5.28 shows that the curve is not linear which is caused by non-linear zener–diode component. The zener–diode is needed to protect the node from over-voltage, the solution to non-linearity is non-linear curve fitting. The curve in Figure 5.28 is

$$y = 212.05 \times ln(x) - 875.56, \tag{5.7}$$

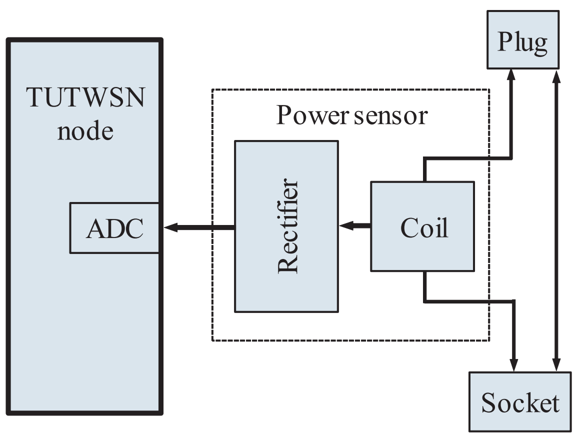

Figure 5.26: *The block diagram of the power sensor.*

Figure 5.27: *The Power Meter Sensor.*

but because the x is the unknown, to get watts from ADC value

$$x = e^{\frac{y+875.56}{212.05}}. \tag{5.8}$$

The equation implementations on software are extremely time and memory consuming but they could be implemented to MCU environment using Taylor polynomials. Because the power values are limited, the non-linear curve is estimated by four separate lines.

The power meter sensor is purely measuring sensor so the packet choice is measurement packet and multipurpose measurement packet. This also dictates data visualization in WSN Control Panel to be measurement visualization.

The power meter sensor application is simple and it only asks power values from the sensor driver. Power meter sensor driver gets ADC value from the ADC of the MCU and calculates power values using the fitted lines.

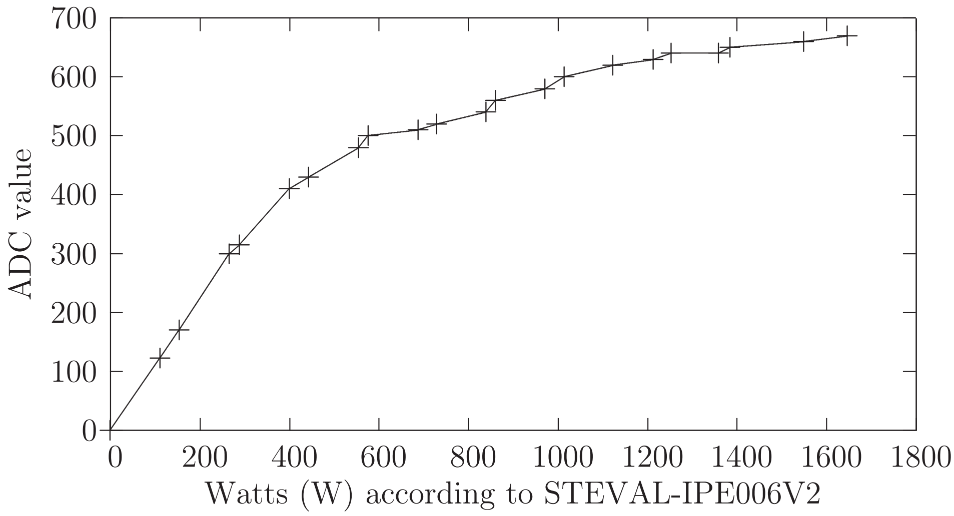

Figure 5.28: *The relation between ADC values of the node and watts.*

Possible sources of error in power measurement are caused by the voltage of the electric grid, frequency and wave form which may vary. According to EN 50160:2007 [13], the frequency has to be between 49.5 Hz and 50.5 Hz 99.5 % of the time. The voltage is 230 V_{rms} but the variation in the voltage should not exceed ± 10 %. The maximum current from the wall socket is 16 amperes, which means that with the maximum current and with the maximum or the minimum voltage, the power value differs 368 watts from power value with nominal voltage of 230 V_{rms}. The accuracy of the power meter sensor is ± 10 % due to the fact that the sensor does not measure voltage.

The power meter sensor is passive because it gets all the energy it needs from the wire, however the power meter sensor needs ADC of the MCU to provide measurements. The energy consumption of one ADC channel is in the range of hundreds of nanojoules and extremely difficult to measure.

The end-to-end testing is done with the same method as in the integration testing step but the data is transferred through the TUTWSN. The power meter sensor is deployed in three WSNs: in a residential WSN, in a real estate surveillance WSN and in a laboratory WSN. The Figure 5.29 presents the power graph received from a TV set in the residential WSN.

The power meter sensor conclusion table is presented in Table 5.5. One notable thing in the power sensor is that it is passive.

The future development of the power sensor looks interesting. The sensor components will be changed and the sensor is going to use pre-calibrated module which eases the sensor integration. Possibility to make Google PowerMeter [22] integration to the backbone system will be evaluated after the power sensor is available.

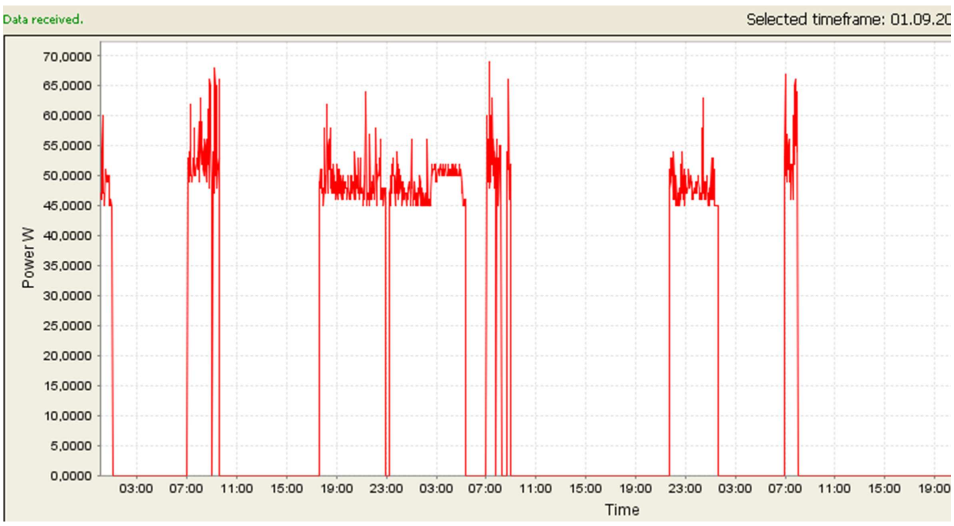

Figure 5.29: *The power meter sensor monitoring power consumption of a TV set in the residential WSN.*

Table 5.5: *The power meter sensor conclusion table*

Memory				Electrical			
Program bytes	**Data bytes**	**Program %**	**Data %**	**Current (mA)**	**Voltage (V)**	**Power (mW)**	**Energy (J)**
1227	12	0.9	0.3	passive	passive	passive	passive

5.4 Radon Sensor

Ionizing radiation is invisible, odorless and tasteless and is therefore impossible to observe by humans. Ionizing radiation is a major health issue and is caused by radioactive decay in the bedrock, on top of which we live. Major cause of ionizing radiation is radon that is an intermediate product in the decay process of uranium [74]. Concentrations of radon are low outdoors but can build up indoors [12] especially in underground structures or when ground is covered with snow. According to United Nations [60] about half of non-medical ionizing radiation exposure is caused by radon-222 . Average personal dose equivalent in Finland is 3.7 mSv and half of that is caused by indoor radon [73]. Also, lung cancers directly caused by radon in Finland is 200 which is 10 percent of all lung cancers in Finland.

Activity of radioactive substance is expressed with nuclear modifications per second. A common unit of it is curie (Ci) which is defined to be 3.70×10^{10} decays per second. The SI-unit of this quantity is becquerel (Bq) and one becquerel is equal to one modification in nucleus. Activity is normally expressed as activity per volume or weight. Indoors it is measured as becquerel per cubic meter (Bq/m^3). [91]

Dose equivalents are measured in sieverts (Sv) or millisieverts (mSv). One sievert equals 1 J/kg

Radiation and Nuclear Safety Authority Finland (STUK) [63] offers radon measurement service that is implemented with disposable cans, seen in Figure 5.30. Recommended amount of these cans is two per house or one per floor and the cans should be in the apartment for two months between November and April. After this two month period the cans are sent and analyzed at STUK which takes up to two months.

According to STUK if radon measurement analysis exceeds 400 Bq/m^3 immediate actions should be taken to reduce the radon concentration. If the analysis shows 200 –400 Bq/m^3 ventilation and other mild actions should be taken and less than 200 Bq/m^3 is acceptable. STUK states that 63 % of lung cancers are caused in radon concentrations less than 200 Bq/m^3 [90].

A long radon measurement period done in already existing building is an ideal situation for the WSN because there is no need for holes or wiring, just deploy and forget. If the radon concentration exceeds the limits and actions are taken to lower

Figure 5.30: *Radon measurement can [90]*

it, measurements can easily be redone to prove that actions have effect.

One method to measure radon is to use gas detectors but this thesis presents a different approach by measuring radioactive decays. This approach is chosen because radon gas sensors are expensive and not widely available. Radon itself is not a dangerous gas because it is noble gas but as it alpha decays it releases helium nucleus which is a very high energy particle and this high energy in cells causes changes. This decay process continues and more alpha and beta decays happen. To measure alpha decays from space, it requires two sensors, one sensor measures all radioactive decays in one place and the second sensor measures only the beta decays. The beta decay readings are reduced from the sensor which measures all radioactive decays and the result is the number of alpha decays which are related to Radon.

Radiation sensor used is RM-60 and it is manufactured by Aware Electronic [16]. The sensor gives pulses proportional to amount of radiation it detects. The RM-60 is basically a geiger counter.

These 'clicks' produced by the RM-60 could be detected by using interrupt pin of MCU but interrupts would affect the normal activity of the node or would cause the loss of some pulses if interrupts would be disabled during critical sections. To prevent using interrupts a special sensor fitting was designed and implemented and its block diagram is presented in Figure 5.31.

Components used in sensor fitter are SN74HC590A [80] 8-bit binary counter by Texas Instruments and Microchip MCP23008 [53] 8-Bit I/O expander with serial I^2C interface. Sensor fitter also has an inverter made of discrete FETs to invert the negative pulses of RM-60 to positive. The principle behind the sensor fitter is to transform serial pulses to parallel and read the number of pulses from the expander using I^2C interface. The picture of the prototype radon sensor is presented in Figure 5.32. The Figure 5.32 has a TUTWSN node, the sensor fitting and the RM-60.

The integration testing step included the electrical testing the senor fitting. After

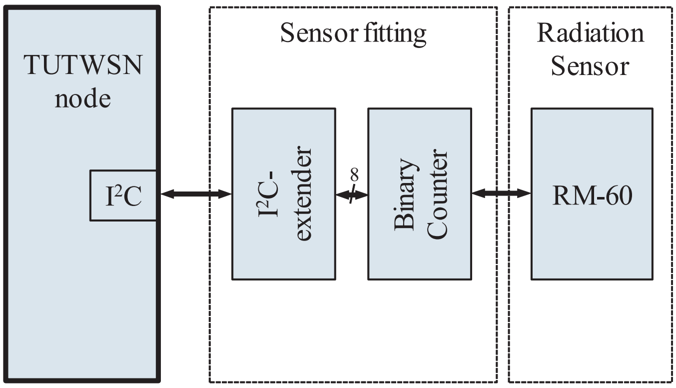

Figure 5.31: *The block diagram of the radon sensor.*

the sensor fitting is tested, RM-60 was connected to it and the sensor was connected to the node. The program module, which uses I^2C of the node to communicate with MCP23008 I^2C extender, was programmed and tested. After the tests proved that counter value increased when RM-60 created a pulse the next step is considered.

The Application Data structure is presented in Figure 5.33 and it contains two values for measurement data: radiation value and pulses per minute. Also, the structure contains sensor type field which can be used to contain the location of the sensor and automate calculation in the NG.

The application for radon calculates the pulses from the last minute. This value

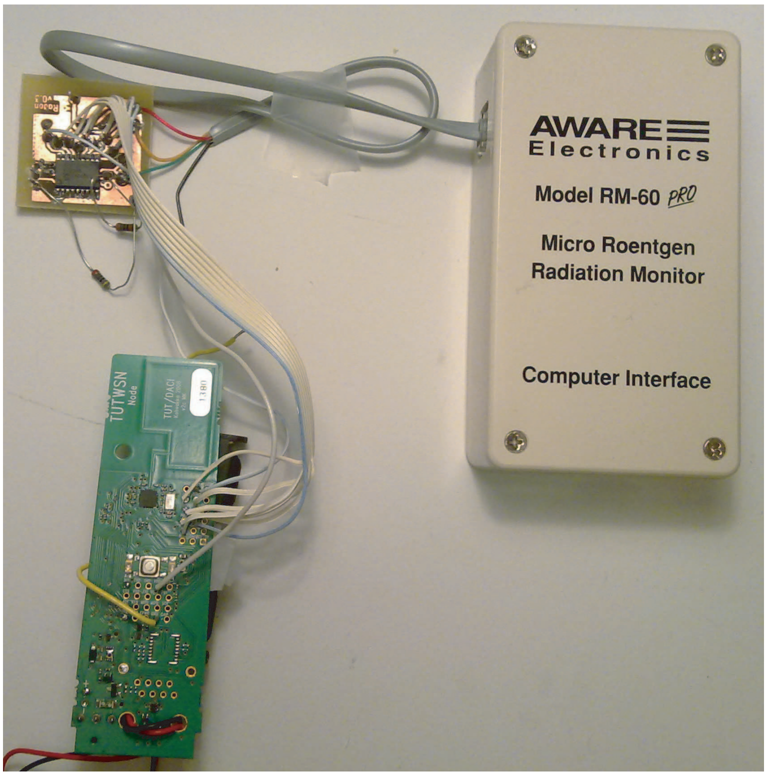

Figure 5.32: *The picture with the TUTWSN node, the sensor fitting and the RM-60.*

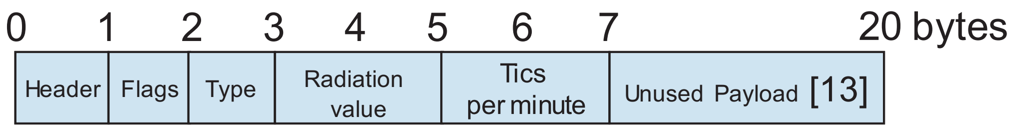

Figure 5.33: *The Radon sensor Application Data structure.*

is used to calculate the radiation value. Both values are put to Application Data structure and transferred to NG when the application is executed. These values are then visualized with WSN Control Panel and calculated radiation is presented next to the node icon.

End-to-end testing proved to be extremely difficult because there is no radioactive radon gas available on the market. One possibility could be to use radioactive ore to calibrate but this would not be the same as radon gas. The sensor is placed in the fourth floor which has very little or no radon in the air and the graph covering ten days is presented in Figure 5.34.

The memory and electrical characteristics of the radon sensor are presented in Table 5.6. From the table it can be observed that the sensor is not battery operable because it uses too much power and it needs to be mains powered. This reduces the number of places the sensors can be placed.

In the future the radon sensor will be made of a cheaper geiger counter but this is only the first step. This step does not solve the fundamental problem of using too

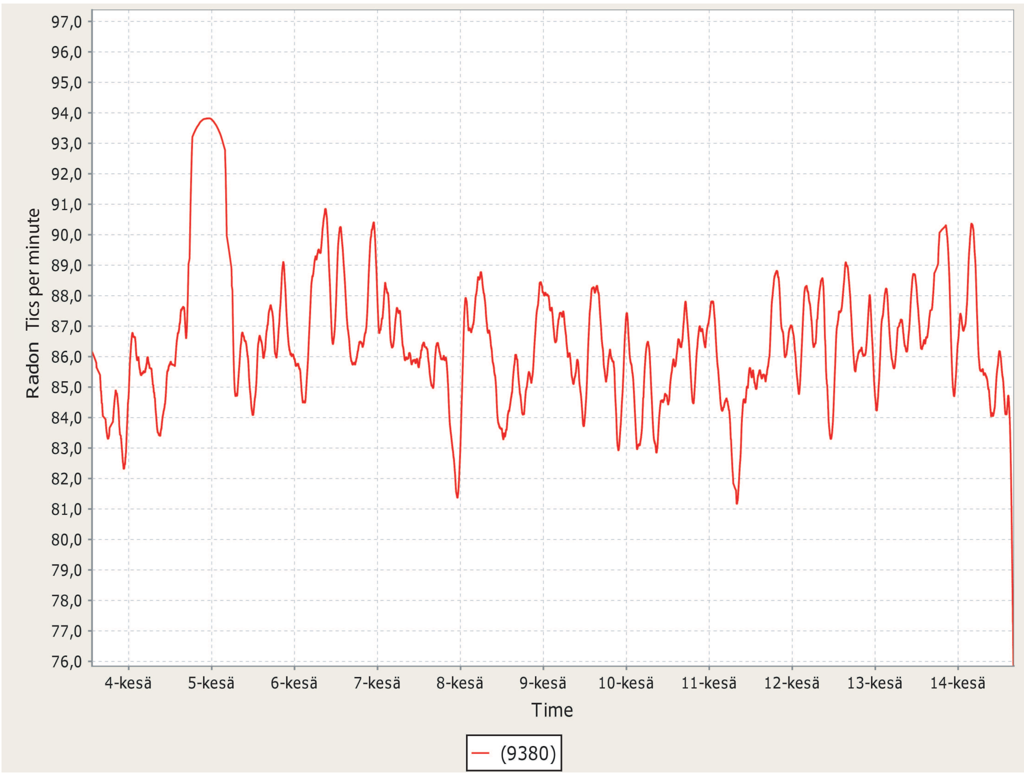

Figure 5.34: *Ten day period of radon measurements.*

Table 5.6: *The radon sensor conclusion table.*

Memory				Electrical			
Program bytes	Data bytes	Program %	Data %	Current (mA)	Voltage (V)	Power (mW)	Energy (J)
1348	8	1.0	0.2	10.0	2.50	25.0	cont.

much power. The next step will be optical radon sensor [89] which optically detects radon gas particles and also consumes less power which makes the sensor battery operable.

5.5 Piezoelectric Motion Detector

Outdoors motion detecting has proven to be a difficult task. Low energy consuming passive infra-red (PIR) sensors generate false positive motion detections in bright and varying sun light conditions because the sun light warms the environment unevenly. [6]. A solution to this problem was derived from the piezoelectric effect.

Some materials, like quartz, generate electric potential when deforming stress is directed to the material, this is called the direct piezoelectric effect. The converse piezoelectric effect happens when a voltage is applied to the material and it deforms [19, p.9]. The direct piezoelectric effect is used in sensors to measure force, torque and acceleration [19, p.7]. The converse piezoelectric effect is used in actuators, and oscillators to generate ultrasound [19, p.6].

All piezoelectric materials are also pyroelectric [88]. The pyroelectric effect means that material generates electric potential difference when it is exposed to temperatures higher or lower to temperature of material and the effect has to be taken into consideration when testing the sensor by making sure that the material is at the same temperature as the environment.

The piezoelectric motion detector consists of a piezo element which is in coaxial cable form and an amplifier PCB. The amplifier is a charge amplifier which converts small charges to voltage. The amplifier is built around an operational amplifier MAX4471 [46]. Coaxial cable form makes it possible to connect the piezo cable like any other coaxial cable. This property is used to connect the piezo cable to normal inactive coaxial cable which is then connected to the amplifier PCB with SMA connector. [6] Inactive coaxial cable is used to extend the piezo cable so that the piezo cable would not be hanging in mid-air and causing false alarms. The block diagram of the piezoelectric motion detector is presented in Figure 5.35. The sensor is connected to interrupt input of the node.

Figure 5.36 presents the prototype PCB on the left and the production model on the right. Less than 20 working prototypes were manufactured between the years 2007 –2010 but as the demand for piezoelectric motion detector will increase in the

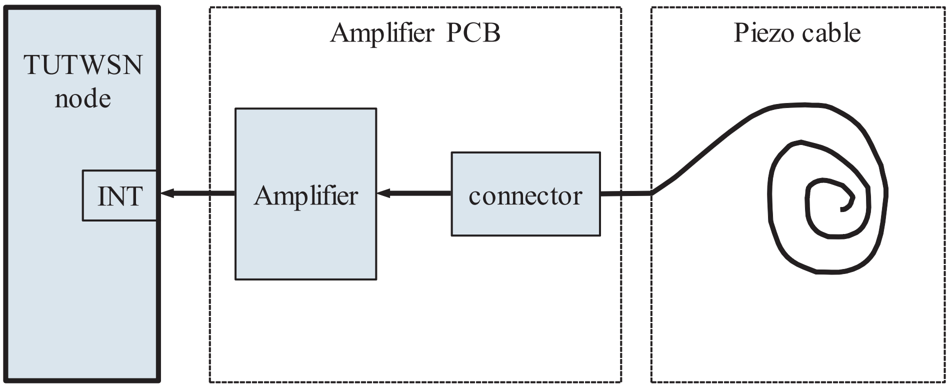

Figure 5.35: *The block diagram of the piezoelectric motion detector.*

future a production model is manufactured to satisfy the growing demand.

The current of the sensor is measured and the results are following: in use the sensor draws 4.0 μA continuously and when the piezo cable is under stress the sensor draws 40.0 μA but this spike lasts only for a short period of time. The conclusion from the power consumption measurements is that the piezoelectric motion detector is usable in battery powered nodes.

The detection range is defined in the integration testing step. The sensing piezo cable can and should be buried underground. The buried cable does not cause false alarms and it is almost impossible to detect. The cable is buried 3 to 5 cm underground. The sensor and the TUTWSN node are presented in Figure 5.37. To define the detection range a person walks next to the piezo cable with a known distance along the piezo cable and the sensor output values are recorded. To record the values, the sensor is connected to one of the ADC inputs of the node and these values are printed through UART to the PC screen and logged.

The data sheet of PIC18LF8722 [52] states that input high voltage has to be

$$V_{HIGH} = 0.8 \times V_{DD} \tag{5.9}$$

Figure 5.36: *On the left a prototype PCB and on the right the manufactured model which can be fitted to the TUTWSN node enclosure extension part*

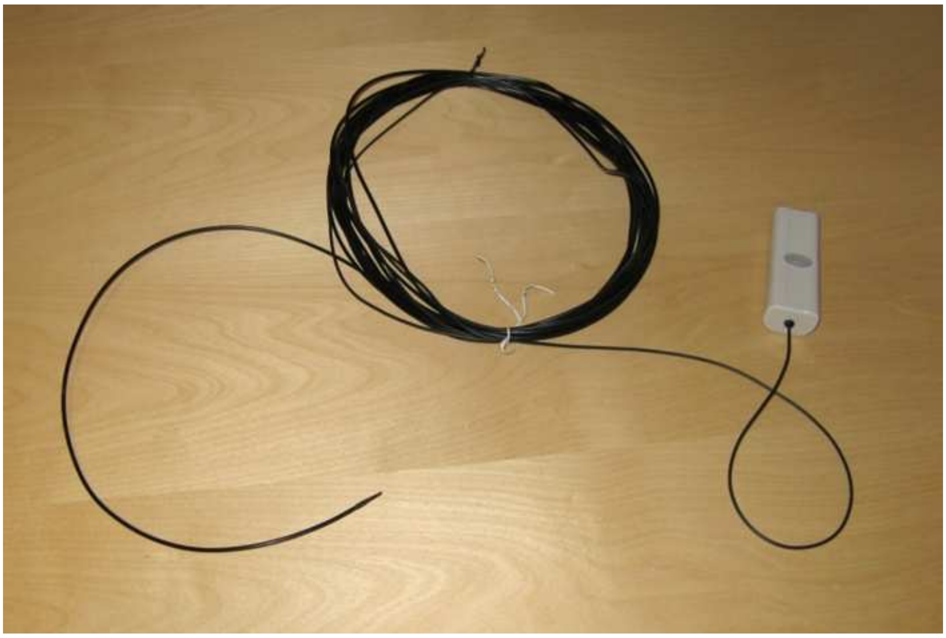

Figure 5.37: *The TUTWSN node and the piezo cable.*

which can be converted to be usable with the ADC to

$$ADC_{value} = 0.8 \times ADC_{range} = 0.8 \times 1023 = 818.4 \approx 818 \qquad (5.10)$$

this means that the ADC value has to be over 818 to trigger an interrupt.

Two most relevant sensor output graphs are presented in Figures 5.39 and 5.40. In the Figure 5.39 the highest AD value spikes are well over 800 but in the Figure 5.40 the highest AD value spikes are not even 700 so the maximum detection distance is defined from the logged values to be 1.00 meters. It is almost impossible to jump over the detection area of the piezo cable because it creates a barrier which is 2.00 meters wide.

The sensor is designed to be motion detection sensor so the sensor is integrated to an existing motion detection application and only modification needed to the node software is to add new bit for piezoelectric motion detector. The motion detection application payload is presented in the Figure 5.38. The visualization in the WSN Control Panel is the alert event and name the new bit to piezo.

The end-to-end testing is done with WSN which has seven piezoelectric motion detectors installed in to a forest. The piezoelectric motion detectors are installed

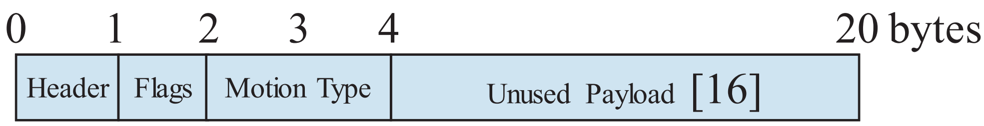

Figure 5.38: *The motion detection application Data Structure.*

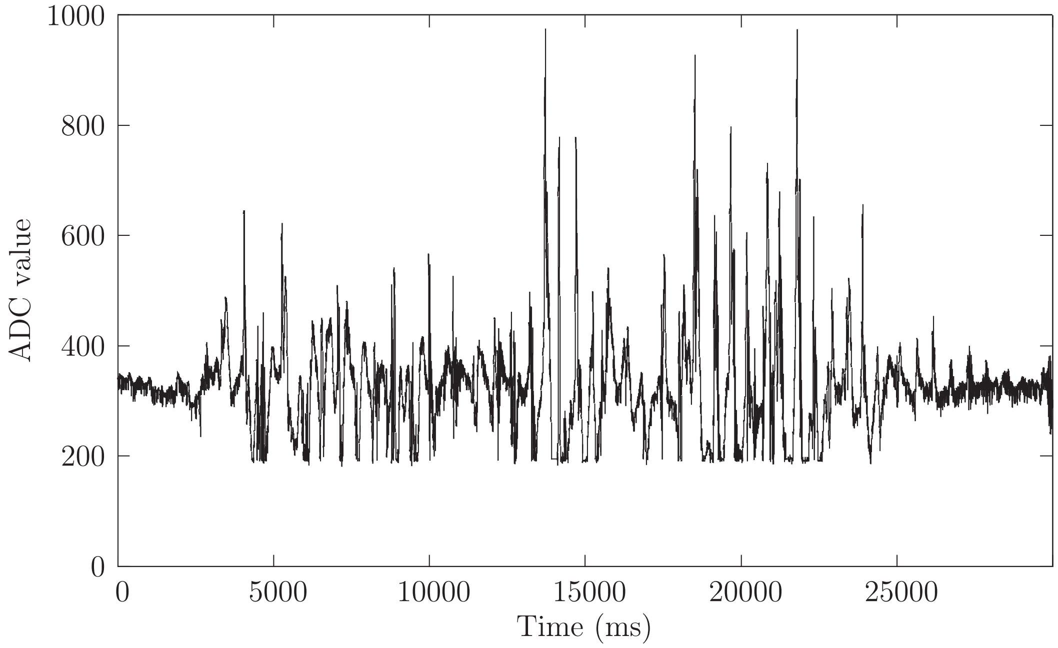

Figure 5.39: *Output of the piezo cable sensor 1.00 meter distance from the cable*

in a grid formation, as seen in Figure 5.41. The piezo cables are presented as grey rectangles. The piezo cables are buried underground in ditches as is done in the integration stepping step. The end-to-end testing comprised of multiple walks across the grid formation and from the WSN measurements it is possible to define the walked path.

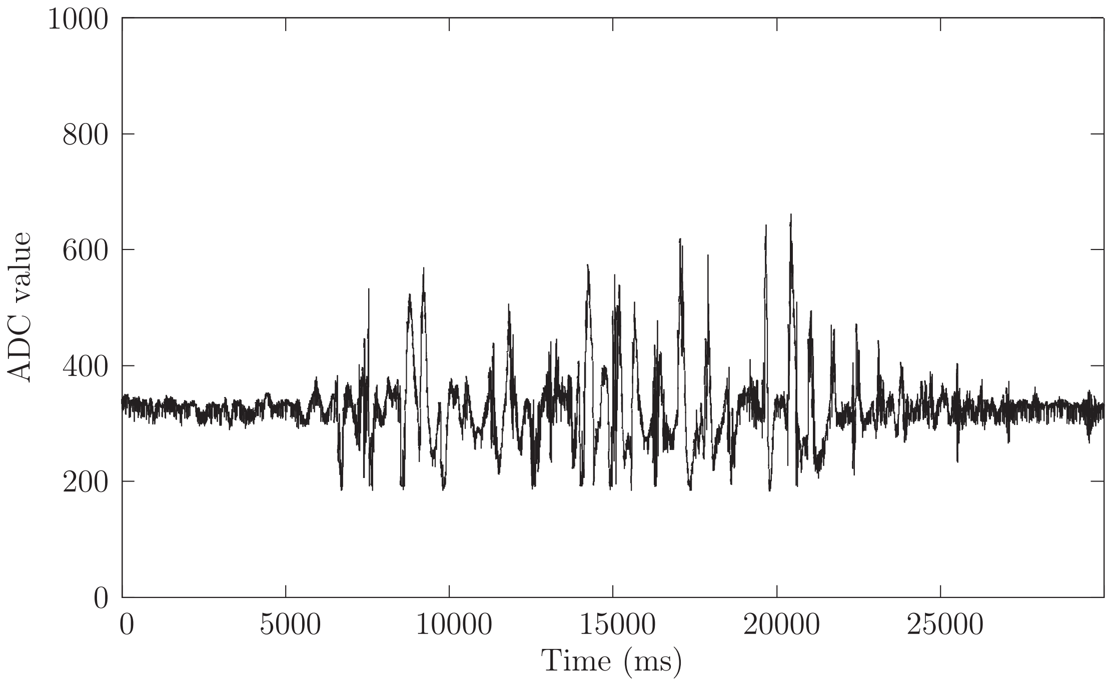

Figure 5.40: *Output of the piezo cable sensor 1.50 meter distance from the cable*

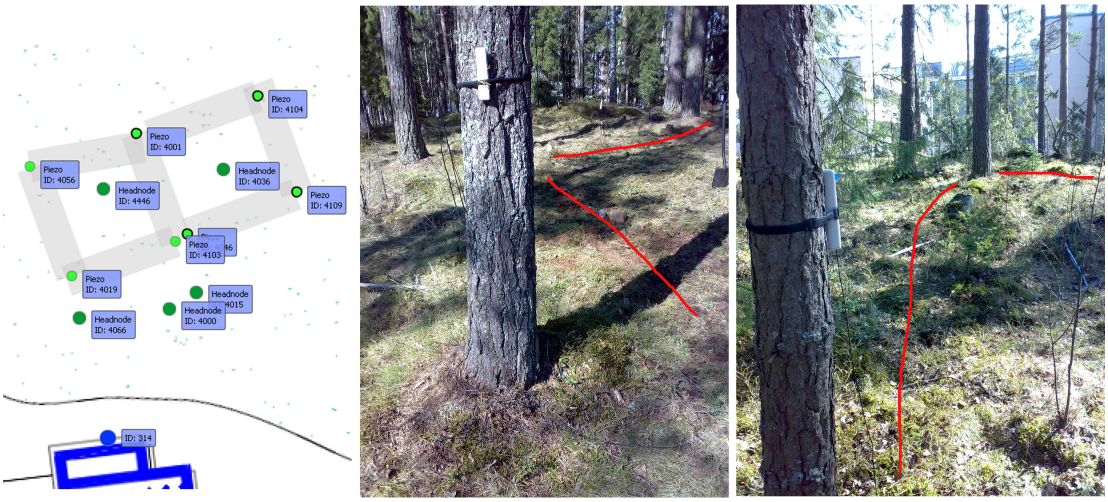

Figure 5.41: *From left to right: WSN Control Panel visualization, node 4019, node 4001. The buried piezo cable is presented as a red line. [6]*

Table 5.7 presents the memory usage and electric characteristics of the piezoelectric motion detector application. The electrical characteristics present the passive case when piezo cable is not under stress. The sensor usable in battery powered node.

The sensor board does not limit the form of usable piezoelectric material but the cable is ideal for outdoor motion detection and so far it is used in multiple locations. It is even possible to install the cable under doorstep or under concrete slab and get motion detections when a person walks by. The cable can also be used indoors under a carpet.

Table 5.7: *The piezo sensor conclusion table*

Memory				Electrical			
Program bytes	**Data bytes**	**Program %**	**Data %**	**Current (μA)**	**Voltage (V)**	**Power (μW)**	**Energy (J)**
1147	29	0.9	0.7	4.0	2.50	10	cont.

6. TIME ANALYSIS OF THE INTEGRATION PROCESS

This chapter analyzes the amount of time it took to integrate the sensors.

The Multipurpose Signal Sensor has three different sensors and one actuator. Selecting the components took three days. The sensor needed PCB which took one week to design and four weeks to get a prototype PCB. The integration to node took one weeks and end-to-end testing took two weeks.

Selecting the air velocity sensor took one day. The delivery of the sensors was four weeks during which an adapter PCB was designed. Integration to node and end-to-end testing took one week.

The component choosing process of the power meter sensor was done during one day. The integration to node took one week which included PCB designing and manufacturing of all three of the existing sensors.

The radon sensor was chosen in one day. Designing the fitter PCB and manufacturing the only existing sensor took one week.

The piezoelectric motion detector was the easiest because piezoelectricity was the only possibility to gain required properties. To get piezo cable, it took a period of two weeks to get the cable. The PCB was designed and prototyped by another department which speeded up the integration to node phase. End-to-end testing lasted the whole summer because the WSN was deployed near the university.

Selecting the sensor took one day in all the integrated sensors but as this thesis proved this sensor probably is to be used in the first version and as the understanding increases the sensor is changed. From selecting the sensor it can take as long as four weeks to receive the first sensors for integration testing. If the sensor is only an element and needs supporting electronics, time must be reserved for PCB design and manufacturing. The PCB design takes one week and additional time must be reserved for PCB prototyping. An industrial scale manufacturing of the PCBs takes months.

Integration testing the sensor phase takes one day if there are no problems. This phase also includes the application data structure design phase.

Integration to server infrastructure phase takes five days and is done in parallel to node integration.

The integration of five sensors enable creating a timetable. The timetable is presented in Figure 6.1 and it helps to estimate how much work integrating a sensor to a WSN requires. Also, it helps time management and helps to estimate the delivery time of a WSN.

The longest time is reserved for "Integration to node" phase because the sensor may require calibration which is not the key issue in the integration process and that is why it is recommended to use pre-calibrated, digital sensors. These sensors do not require calibration and focus can be kept in the integration, not the calibration.

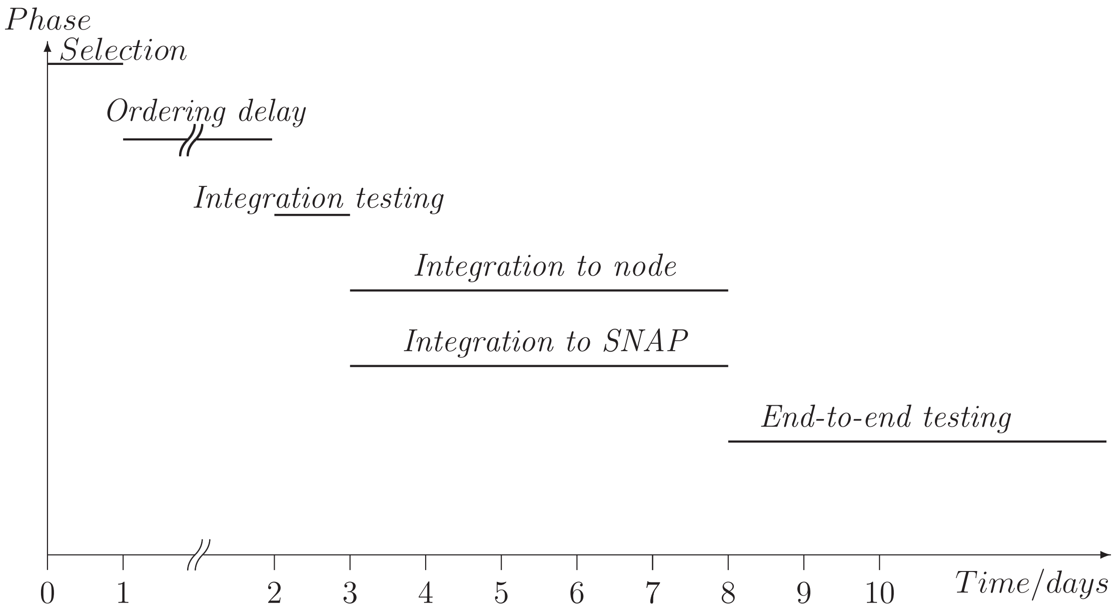

Figure 6.1: *Sensor integration process time table.*

7. CONCLUSIONS

Objectives of this thesis are met and the new integration process developed and documented. The result is that it is now possible to estimate work hours needed to integrate a new sensor. The documentation helps anyone to integrate new sensors to TUTWSN and to use the same process in some other WSN.

Two new ideas came forward during the work. Every sensor requires an own application which are quite similar with initialization, measuring, and application data structure filling. The application layer would consume far less memory if it had only one application which would implement and utilize all of the sensor drivers and handle application Data structures.

The second new idea is related to the energy consumption measurement of the sensor. In this new approach, a capacitor would produce the current required by the node but the recharging would be automated with a comparator circuit. The circuit would provide pulses in relation to the current drawn by the node and by counting the pulses it would be possible to measure the energy consumption of the node.

Challenges in sensor integration are in the lowest software level which always has to be programmed individually to every sensor.

Future work will include searching and following the development of plug-and-play sensors which could be the solution by defining an architecture with standardized electrical interface to the node or a self-identification protocol, allowing node and sensor to negotiate and sensor describe itself to the node.

External, plug-and-play sensors of TUTWSN will have I^2C memory chip to identify the sensor, much like the TEDS of IEEE 1451 Smart Transducer Interface Standard. When a sensor is connected, the node identifies the sensor and requests new sensor application through the WSN. This application is then transferred to the node and node starts using the new application.

It would be interesting to use the sensor integration process to integrate sensors to other WSNs and evaluate the feasibility of the process.

REFERENCES

[1] I.F. Akyildiz, Weilian Su, Y. Sankarasubramaniam, and E. Cayirci. A survey on sensor networks. *Communications Magazine, IEEE*, 40(8):102–114, Aug 2002.

[2] Geoff W. Allen, Konrad Lorincz, Jeff Johnson, Jonathan Lees, and Matt Welsh. "Fidelity and yield in a volcano monitoring sensor network". In *OSDI '06: Proceedings of the 7th USENIX Symposium on Operating Systems Design and Implementation*, page 27, Berkeley, CA, USA, 2006. USENIX Association.

[3] ZigBee Alliance. *"ZigBee Smart Energy Profile Specification"*. 2009.

[4] ZigBee Alliance. *"ZigBee Specification"*. 2009.

[5] Analog Devices, Inc. ADS5301 data sheet. Available: `http://www.analog.com/static/imported-files/data_sheets/AD5301_5311_5321.pdf`, 2007.

[6] Jani Arvola. *Liikkeen Havaitseminen Pietsokaapelilla*. BSc thesis, 2008. Tampere University of Technology (TUT), Tampere.

[7] R. Beckwith, D. Teibel, and P. Bowen. "Report from the field: results from an agricultural wireless sensor network". In *Local Computer Networks, 2004. 29th Annual IEEE International Conference on*, page 471–478, Nov. 2004.

[8] Alberto Cerpa, Jeremy Elson, Deborah Estrin, Lewis Girod, Michael Hamilton, and Jerry Zhao. "Habitat monitoring: application driver for wireless communications technology". volume 31, page 20–41, New York, NY, USA, 2001. ACM.

[9] G Comte-Bellot. Hot-wire anemometry. *Annual Review of Fluid Mechanics*, 8:209–231, Aug 1976.

[10] MEMSIC Corporation. Website. Available: `http://www.memsic.com`.

[11] Crossbow Corporation. MTS/MDA Sensor Board Users Manual. Available: `http://cone.informatik.uni-freiburg.de/teaching/praktikum/Adhocnetworks-w07/wsn/mica2-cd/Manuals%20and%20Docs/MTS-MDA_Series_Users_Manual_7430-0020-04_B.pdf`, 2006.

[12] S. Darby, D. Hill, H. Deo, A. Auvinen, J. M. Barros-Dios, H. Baysson, F. Bochicchio, R. Falk, S. Farchi, A. Figueiras, M. Hakama, I. Heid, N. Hunter, L. Kreienbrock, M. Kreuzer, F. Lagarde, I. Mäkeläinen, C. Muirhead, W. Oberaigner, G. Pershagen, E. Ruosteenoja, A. S. Rosario, M. Tirmarche, L. Tomásek, E. Whitley, H. E. Wichmann, and R. Doll. "Residential

radon and lung cancer--detailed results of a collaborative analysis of individual data on 7148 persons with lung cancer and 14,208 persons without lung cancer from 13 epidemiologic studies in Europe.''. *Scand J Work Environ Health*, 32 Suppl 1:1–83, 2006.

[13] Comité Européen de Normalisation Electrotechnique CENELEC. *''EN 50160:2007 Voltage characteristics of electrity supplied by public distribution networks''*.

[14] M. Dunbar. ''Plug-and-Play Sensors in Wireless Networks''. *Instrumentation Measurement Magazine, IEEE*, 4(1):19–23, mar 2001.

[15] E+E Elektronik Ges.m.b.H. EE575 miniature air velocity transmitter. Available: `http://www.epluse.com/uploads/tx_EplusEprDownloads/datasheet_EE575_e_03.pdf`, 2009.

[16] Aware Electronics. Website. Available: `http://www.aw-el.com`.

[17] Elisa, Oyj. Elisa Vahti service. Available: `http://www.elisa.fi/vahti/`, 2010.

[18] HART Communication Foundation. ''WirelessHART Technical Data Sheet''. Available: `http://www.hartcomm.org/`.

[19] Gustav Gautschi. *Piezoelectric Sensorics: Force, Strain, Pressure, Acceleration and Acoustic Emission Sensors, Materials and Amplifiers*. Springer, 2006.

[20] Jukka Haaramo. *Delivery Process of Wireless Sensor Networks*. MSc thesis, 69 pages, 2010. Tampere University of Technology (TUT), Tampere.

[21] Aki Hakkarainen. *Antenna Design on Wireless Sensor Networks*. MSc thesis, 69 pages, 2007. Tampere University of Technology (TUT), Tampere.

[22] Google Inc. Website. Available: `http://www.google.com/powermeter`.

[23] Honeywell Inc. Honeywell TEDS website. Available: `http://www.sensotec.com/pnpterms.shtml`.

[24] National Instruments Inc. National Instruments TEDS website. Available: `http://www.ni.com/teds`.

[25] Softnoze USA Inc. Website. Available: `http://www.softnoze.com`.

[26] Texas Instruments. ''CC2430ZDK ZigBee Development Kit Pro''. Available: `http://focus.ti.com/lit/ug/swrd006a/swrd006a.pdf`.

[27] Tomi Jäntti. *Server Platform for Wireless Sensor Network Applications*. MSc thesis, 69 pages, 2010. Tampere University of Technology (TUT), Tampere.

[28] J. M. Kahn, R. H. Katz, and K. S. J. Pister. "Next century challenges: mobile networking for "Smart Dust"". In *MobiCom '99: Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, page 271–278, New York, NY, USA, 1999. ACM.

[29] M. Kohvakka, M. Hannikainen, and T.D. Hamalainen. "Wireless sensor prototype platform". In *Industrial Electronics Society, 2003. IECON '03. The 29th Annual Conference of the IEEE*, volume 2, page 1499–1504 Vol.2, Nov. 2003.

[30] M. Kohvakka, M. Hannikainen, and T.D. Hamalainen. "Energy optimized beacon transmission rate in a wireless sensor network". In *Personal, Indoor and Mobile Radio Communications, 2005. PIMRC 2005. IEEE 16th International Symposium on*, volume 2, page 1269–1273 Vol. 2, Sept. 2005.

[31] Mikko Kohvakka. *Medium Access Control and Hardware Prototype Designs for Low-energy Wireless Sensor Networks*. PhD thesis, Tampere University of Technology (TUT), Tampere, 2009.

[32] Mikko Kohvakka, Jukka Suhonen, Mauri Kuorilehto, Ville Kaseva, Marko Hännikäinen, and Timo D. Hämäläinen. "Energy-efficient neighbor discovery protocol for mobile wireless sensor networks". *Ad Hoc Netw.*, 7(1):24–41, 2009.

[33] Lakshman Krishnamurthy, Robert Adler, Phil Buonadonna, Jasmeet Chhabra, Mick Flanigan, Nandakishore Kushalnagar, Lama Nachman, and Mark Yarvis. "Design and Deployment of Industrial Sensor Networks: Experiences from a Semiconductor Plant and the North Sea". In *SenSys '05: Proceedings of the 3rd international conference on Embedded networked sensor systems*, page 64–75, New York, NY, USA, 2005. ACM Press.

[34] M. Kuorilehto, M. Hannikainen, and T. Hamalainen. Rapid design and evaluation framework for wireless sensor networks. *Ad Hoc Networks*, 6(6):909–935, August 2008.

[35] Mauri Kuorilehto. *System Level Design Issues in Low-Power Wireless Sensor Networks*. PhD thesis, Tampere University of Technology (TUT), Tampere, 2008.

[36] Mauri Kuorilehto, Mikko Kohvakka, Jukka Suhonen, Panu Hämäläinen, Marko Hännikäinen, and Timo D. Hämäläinen. *"Ultra-Low Energy Wireless Sensor Networks in Practice"*. 2007.

[37] Kang Lee. "IEEE 1451: A Standard in Support of Smart Transducer Networking". In *Instrumentation and Measurement Technology Conference, 2000. IMTC 2000. Proceedings of the 17th IEEE*, volume 2, page 525–528 vol.2, 2000.

[38] Libelium Comunicaciones Distribuidas S.L. Agriculture Board Technical Guide. Available: `http://www.libelium.com/documentation/waspmote/agriculture-sensor-board_eng.pdf`, 2010.

[39] Libelium Comunicaciones Distribuidas S.L. Events Board Technical Guide. Available: `http://www.libelium.com/documentation/waspmote/events-sensor-board_eng.pdf`, 2010.

[40] Libelium Comunicaciones Distribuidas S.L. Gases Board Technical Guide. Available: `http://www.libelium.com/documentation/waspmote/gases-sensor-board_eng.pdf`, 2010.

[41] Controlmatic Oy Ltd. Website. Available: `http://www.controlmatic.fi`.

[42] Sensinode Ltd. Website. Available: `http://www.sensinode.com`.

[43] Wirepas Ltd. Website. Available: `http://www.wirepas.com`.

[44] Alan Mainwaring, David Culler, Joseph Polastre, Robert Szewczyk, and John Anderson. "Wireless sensor networks for habitat monitoring". In *WSNA '02: Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, page 88–97, New York, NY, USA, 2002. ACM.

[45] Maxim Integrated Products. MAX8880/MAX8881 12V, Ultra-low-IQ, low-dropout linear regulators with pok. Available: `http://datasheets.maxim-ic.com/en/ds/MAX8880-MAX8881.pdf`, 2007.

[46] Maxim Integrated Products. MAX4471 Single Rail-to-Rail Op Amp. Available: `http://datasheets.maxim-ic.com/en/ds/MAX4464-MAX4474.pdf`, 2010.

[47] MEMSIC Corporation. MICA2 Datasheet. Available: `http://www.memsic.com/support/documentation/wireless-sensor-networks/category/7-datasheets.html?download=147%3Amica2`, 2010.

[48] MEMSIC Corporation. MTS300/310 Data Sheet. Available: `http://www.memsic.com/support/documentation/wireless-sensor-networks/category/7-datasheets.html?download=168%3Amts-mda`, 2010.

[49] MEMSIC Corporation. MTS400/420 Data Sheet. Available: `http://www.memsic.com/support/documentation/wireless-sensor-networks/category/7-datasheets.html?download=174%3Amts400_420`, 2010.

[50] MEMSIC Corporation. TelosB Datasheet. Available: http://www.memsic.com/support/documentation/wireless-sensor-networks/category/7-datasheets.html?download=152%3Atelosb, 2010.

[51] MeshNetics. "ZigBit Development Kit". Available: http://www.meshnetics.com/netcat_files/Image/S-ZDK-451-%28ZigBit%20Development%20Kit%20Users%20Guide%29.pdf.

[52] Microchip Technology, Inc. PIC18F8722 family data sheet. Available: http://ww1.microchip.com/downloads/en/DeviceDoc/39646b.pdf, 2004.

[53] Microchip Technology, Inc. MCP23008 data sheet. Available: http://ww1.microchip.com/downloads/en/DeviceDoc/21919e.pdf, 2007.

[54] D. Niculescu. Communication paradigms for sensor networks. *Communications Magazine, IEEE*, 43(3):116–122, march 2005.

[55] Nordic Semiconductor ASA. Single chip 433/868/915 MHz transceiver nRF905. Available: http://www.nordicsemi.no/files/Product/data_sheet/nRF905_rev1_4.pdf, 2006.

[56] Nordic Semiconductor ASA. Single chip 2.4 GHz Transceiver nRF24L01+. Available: http://www.nordicsemi.no/files/Product/data_sheet/nRF24L01P_Product_Specification_1_0.pdf, 2009.

[57] International Society of Automation. "ISA-100.11a-2009 Wireless systems for i 5 ndustrial automation: Process control and related applications". Available: http://www.isa.org/ISA100-11a.

[58] National Institute of Standards and Technology. *"NIST Framework and Roadmap for Smart Grid Interoperability Standards Release 1.0 (Draft) "*.

[59] National Institute of Standards and Technology. "IEEE Standard for a Smart Transducer Interface for Sensors and Actuators - Transducer to Microprocessor Communication Protocols and Transducer Electronic Data Sheet (TEDS) Formats". *IEEE Std 1451.2-1997*, 1998.

[60] United Nations Scientific Committee on the Effects of Atomic Radiation. "Sources and Effects of Ionizing Radiation". In *UNSCEAR 2000 Report to the General Assembly with Scientific Annexes. Vol 1: Sources*, page 88–97, New York, USA, 2000. ACM.

[61] PermaSense project. PermaSense homepage. Available: http://www.permasense.ch, 2010.

[62] Arttu Pulli. *Soil Moisture Measurement Application for TUTWSN Wireless Sensor Network*. BSc thesis, 2009. Tampere University of Technology (TUT), Tampere.

[63] Radiation and Nuclear Safety Authority Finland. Website. Available: `http://www.stuk.fi`.

[64] Shockfish SA. Website. Available: `http://www.shockfish.com`.

[65] Schmidt Technology. SS20.260 flow sensor. Available: `http://www.schmidttechnology.de/de/sensorik/download/datenblaetter/SS_20.260_dt-en.pdf`, 2009.

[66] Curt Schurgers and Mani B. Srivastava. "Energy Efficient Routing In Wireless Sensor Networks". In *Military Communications and Networking Conference*, page 350–355, 2001.

[67] P. Sikka, P. Corke, and L. Overs. "Wireless sensor devices for animal tracking and control". In *Local Computer Networks, 2004. 29th Annual IEEE International Conference on*, page 446–454, 2004.

[68] Pavan Sikka, Peter Corke, Philip Valencia, Christopher Crossman, Dave Swain, and Greg Bishop-Hurley. "Wireless adhoc sensor and actuator networks on the farm". In *IPSN '06: Proceedings of the 5th international conference on Information processing in sensor networks*, page 492–499, New York, NY, USA, 2006. ACM.

[69] Gyula Simon, Miklós Maróti, Ákos Lédeczi, György Balogh, Branislav Kusy, András Nádas, Gábor Pap, János Sallai, and Ken Frampton. "Sensor network-based countersniper system". In *SenSys '04: Proceedings of the 2nd international conference on Embedded networked sensor systems*, page 1–12, New York, NY, USA, 2004. ACM.

[70] Libelium Comunicaciones Distribuidas S.L. Website. Available: `http://www.libelium.com`.

[71] IEEE Computer Society. *"Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs)"*. 2006.

[72] STMicroelectronics. STEVAL-IPE006V2 High end meter Demonstration board. Available: `http://www.st.com/stonline/products/literature/ds/12558.pdf`, 2006.

[73] STUK. *"Onko kodissasi radonia?"*. http://www.stuk.fi/julkaisut/katsaukset/pdf/onko_kodissasi_radonia.pdf.

[74] STUK. *"Sisäilman radon"*. http://sateilyturvakeskus.fi/sateilytietoa/sateily_ymparistossa/radon/fi_FI/radon/_files/81558071858823342/default/sisailman_radon_katsaus_lokakuu2008.pdf.

[75] J. Suhonen, M. Kuorilehto, M. Hannikainen, and T.D. Hamalainen. "Cost-Aware Dynamic Routing Protocol for Wireless Sensor Networks - Design and Prototype Experiments". In *Personal, Indoor and Mobile Radio Communications, 2006 IEEE 17th International Symposium on*, page 1–5, Sept. 2006.

[76] Jukka Suhonen, Mikko Kohvakka, Marko Hännikäinen, and Timo D. Hämäläinen. "Design, Implementation, and Experiments on Outdoor Deployment of Wireless Sensor Network for Environmental Monitoring". In *Proc. Embedded Computer Systems: Architectures, Modeling, and Simulation, Samos, Greece, July 2006*, page 109–121, New York, NY, USA, 2006. Springer Berlin / Heidelberg.

[77] Igor Talzi, Andreas Hasler, Stephan Gruber, and Christian Tschudin. "PermaSense: investigating permafrost with a WSN in the Swiss Alps". In *EmNets '07: Proceedings of the 4th workshop on Embedded networked sensors*, page 8–12, New York, NY, USA, 2007. ACM.

[78] FORCE Technology. Website. Available: http://www.forcetechnology.se.

[79] TEKES. TEKES Nordic WSN Market. Available: http://akseli.tekes.fi/opencms/opencms/OhjelmaPortaali/ohjelmat/Ubicom/fi/Dokumenttiarkisto/Viestinta_ja_aktivointi/Seminaarit/Hyoety_sensoriverkoista_20100621/TekesxWSNxmyyntikatsaus.pdf.

[80] Texas Instruments, Inc. SN74HC590A data sheet. Available: http://focus.ti.com/lit/ds/symlink/sn74hc590a.pdf, 2003.

[81] Texas Instruments, Inc. ADS1115 data sheet. Available: http://focus.ti.com/lit/ds/symlink/ads1115.pdf, 2009.

[82] TinyNode. Tinynode standard extension board. Available: http://www.tinynode.com/uploads/media/StandardExtentsionBoard_rev12_01.pdf.

[83] TinyNode. Website. Available: http://www.tinynode.com.

[84] VACUUMSCHMELZE. Website. Available: http://www.vacuumschmelze.de.

[85] VACUUMSCHMELZE. Current Transformer. Available: `http://www.vacuumschmelze.de/fileadmin/documents/pdf/kbproduktblaetter/4622-X503.pdf`, 2003.

[86] Tim Wark, Peter Corke, Pavan Sikka, Lasse Klingbeil, Ying Guo, Chris Crossman, Phil Valencia, Dave Swain, and Greg Bishop-Hurley. "Transforming Agriculture through Pervasive Wireless Sensor Networks". volume 6, page 50–57, Los Alamitos, CA, USA, 2007. IEEE Computer Society.

[87] Geoffrey Werner-Allen, Konrad Lorincz, Matt Welsh, Omar Marcillo, Jeff Johnson, Mario Ruiz, and Jonathan Lees. "Deploying a Wireless Sensor Network on an Active Volcano". volume 10, page 18–25, Piscataway, NJ, USA, March 2006. IEEE Educational Activities Department.

[88] R W Whatmore. "Pyroelectric Devices And Materials". *Reports on Progress in Physics*, 49(12):1335–1386, Dec 1986.

[89] S. Yamamoto, Y. Yoshida, and T. Iida. Development of an underground radon detector using an optical fiber. *Nuclear Science, IEEE Transactions on*, 50(4):987–990, aug. 2003.

[90] YLE. *"Uudet suositukset asuntojen radonista"*. `http://yle.fi/uutiset/kotimaa/2009/09/uudet_suositukset_asuntojen_radonista_1003724.html`.

[91] Hugh D. Young and Roger A. Freedman. *University Physics with Modern Physics*. Addison-Wesley, 2004.